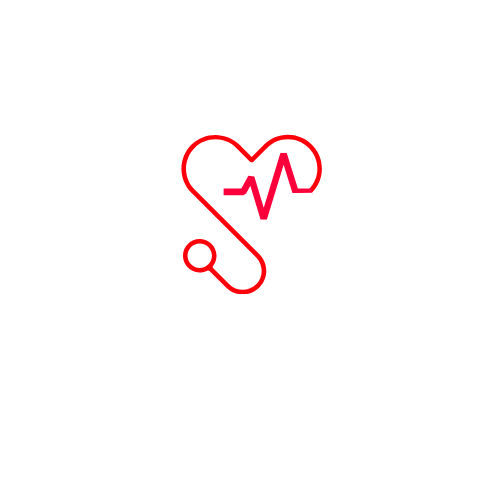
**Final Year Project 2025**

****

**Smart Vitals**

**A Portable Health Monitoring System**

**Submitted by**

**Soya Shrestha**

**(77356846)**

**BSC (Hons) Computing**

**Word Count:**

**Abstract**

In todays’ modern technologically driven world, Smart Vitals presents itself as an IoT enabled, multifunctional and portable device for health tracking. Designed to provide users with an affordable solution replacing traditional tools, it allows people to get insights on their health and surroundings with a single device. The main focus of developing this device is to make health readings accessible to people wherever they are. It not only saves valuable time of the users but also allows them to take preventive measures. Smart Vitals aims to promote health sector as it provides all-in-one solution for monitoring basic vitals. While Smart Vitals is still a prototype, with further enhancement, proper research and refining, it has a promising scope.

**Table of Contents**

[**CHAPTER 1: INTRODUCTION** 6](#_Toc197264116)

[**CHAPTER 2: REVIEW OF LITERATURE** 8](#_Toc197264117)

[**CHAPTER 3: REVIEW OF TECHNOLOGY** 11](#_Toc197264118)

[**3.1 Hardware Components** 11](#_Toc197264119)

[**3.2 Backend** 18](#_Toc197264120)

[**3.3 Operating System** 18](#_Toc197264121)

[**3.4 Language, framework and Libraries** 19](#_Toc197264122)

[**3.5 Version Control** 20](#_Toc197264123)

[**3.6 Documentation** 20](#_Toc197264124)

[**3.7 Web Browser** 20](#_Toc197264125)

[**CHAPTER 4: METHODOLOGY** 21](#_Toc197264126)

[**CHAPTER 5: PRODUCT DESIGN** 23](#_Toc197264127)

[**5.1 General Overview of the system** 23](#_Toc197264128)

[**5.2 Data Flow Diagram** 25](#_Toc197264129)

[**CHAPTER 6: SOFTWARE REQUIREMENT ANALYSIS** 26](#_Toc197264130)

[**CHAPTER 7: IMPLEMENTATION AND TESTING** 27](#_Toc197264131)

[**CHAPTER 8: PRODUCT EVALUATION** 34](#_Toc197264132)

[**8.1 For Device** 34](#_Toc197264133)

[**8.2 For Mobile Application** 35](#_Toc197264134)

[**CHAPTER 9: PROJECT EVALUATION** 36](#_Toc197264135)

[**9.1 Task Sheet** 36](#_Toc197264136)

[**9.2 Gnatt Chart** 37](#_Toc197264137)

[**9.3 Timeline** 38](#_Toc197264138)

[**9.4 Resource Sheet** 39](#_Toc197264139)

[**9.5 GitHub** 40](#_Toc197264140)

[**9.6 Entity Relationship Diagram (ERD) and Composite ERD** 40](#_Toc197264141)

[**9.8 Requirement Catalogue** 43](#_Toc197264142)

[**9.9 Installation Guidelines** 44](#_Toc197264143)

[**9.9.1 Firebase** 44](#_Toc197264144)

[**9.9.2 Arduino IDE** 48](#_Toc197264145)

[**9.9.3 VS code** 50](#_Toc197264146)

[**9.10 Use Case Diagram** 55](#_Toc197264147)

[**CHAPTER 10: SUMMARY AND CONCLUSION** 56](#_Toc197264148)

[**BIBLIOGRAPHY** 59](#_Toc197264149)

**Table of Figures**

[**Figure 1: ESP 32 WROVER TYPE B 11**](#_Toc197264150)

[**Figure 2: MAX30102 12**](#_Toc197264151)

[**Figure 3: AD8232 12**](#_Toc197264152)

[**Figure 4: 10K NTC Thermistor temperature sensor 12**](#_Toc197264153)

[**Figure 5: Resistor 13**](#_Toc197264154)

[**Figure 6: Push Button 13**](#_Toc197264155)

[**Figure 7: OLED 13**](#_Toc197264156)

[**Figure 8: Battery 14**](#_Toc197264157)

[**Figure 9: MQ-135 14**](#_Toc197264158)

[**Figure 10: NEO-6M GPS Sensor 14**](#_Toc197264159)

[**Figure 11: DHT22 15**](#_Toc197264160)

[**Figure 12: Matrix Board 15**](#_Toc197264161)

[**Figure 13: Soldering iron, wax, wire, steel wire wool 15**](#_Toc197264162)

[**Figure 14: Glue Gun 16**](#_Toc197264163)

[**Figure 15: Connecting wire 16**](#_Toc197264164)

[**Figure 16: Creality Ender 3 17**](#_Toc197264165)

[**Figure 17: Power bank module 17**](#_Toc197264166)

[**Figure 18: Reset Button 18**](#_Toc197264167)

[**Figure 19: Type B cable 18**](#_Toc197264168)

[**Figure 20: Smart Vitals 23**](#_Toc197264169)

[**Figure 21: Wireframe for Smart Vitals 24**](#_Toc197264170)

[**Figure 22: Dataflow diagram for Smart Vitals 25**](#_Toc197264171)

[**Figure 23: Task Sheet 36**](#_Toc197264172)

[**Figure 24: Gnatt Chart 37**](#_Toc197264173)

[**Figure 25: Project Timeline 38**](#_Toc197264174)

[**Figure 26: Resource Sheet 39**](#_Toc197264175)

[**Figure 27: GitHub 40**](#_Toc197264176)

[**Figure 28: Entity Relationship Diagram (ERD) 40**](#_Toc197264177)

[**Figure 29: Composite ERD 41**](#_Toc197264178)

[**Figure 30: UML for Smart Vitals 42**](#_Toc197264179)

[**Figure 31: User Use Case Diagram (1) 55**](#_Toc197264180)

[**Figure 32: User Use Case Diagram (2) 55**](#_Toc197264181)

# **CHAPTER 1: INTRODUCTION**

Smart Vital is a portable health monitoring device. It is designed to measure vital signs of an individual. It includes Sp0₂ level, heart rate, ECG, temperature, and blood pressure measurement by utilizing embedded system and IoT. The device provides easy and quick access to essential health parameters in real-time through locally as well as through mobile application. The main purpose for the development of the device is to facilitate users with a reliable and user-friendly product for the healthcare management system.

The device is developed so that it can be placed at first aid or medical box at home or it can be used while travelling as it is compact and easy to carry. A single compact device is used unlike having different devices for measuring health vitals. It offering all in one general health measurement, making sure that you have all the necessary health information well within access whether you are on the go or at home. The device Smart Vitals fulfils the increase in the need and demand of the health care system as it is portable and convenient to use. The system uses ESP32 WROVER as microcontroller, which is the main processing segment for collecting the data from sensors like Max30100, AD8232, 10K NTC Thermistor temperature sensor. The readings from the sensor are locally access via OLED display and also transmitted to mobile application made via Flutter. If there is any abnormality in the reading, alert is sent, ensuring timely intervention.

Smart Vitals was compared with various existing products, during which the uniqueness of the smart vitals was discovered. Additionally, the key features of the smart vitals were also highlighted. All the hardware and software components were well-identified, even the smallest of the components. Without the use of those components the working of smart vitals would lack behind. Smart Vitals followed the agile methodology for its development as it allows flexibility and phased development. The product design includes the overall flow of the system as well as the flow of users interacts with the device andsystem. The featuresof smartvitals with the level of priority was emphasized. After the product was developed, the product was tested continuously finalizing the products’ evaluation ensuring that the system passed every test for both device and mobile application. MS Project was utilized to keep track of the progress of the project with proper planned layout, while GitHub was used for version control and to keep track of the changes made in the project. Similarly, ERD was used to get knowledge on the data structure and better understanding of the database model. At last, the overall project summary and outcome was discussed, including the performance of the system and how the system could be further enhanced for future works.

# **CHAPTER 2: REVIEW OF LITERATURE**

The continuous advancement in technologies in health care has led to advanced approaches like Smart Vitals. The importance of products like Smart Vitals has emerged as a necessary device in healthcare and management due to the rapid global interest for its portability and remote health monitoring features. The incorporation of IoT and micro-controller enable the device to provide users with essential health vitals giving a proper and prompt insights into a person's health state providing assurance to the user. By gaining knowledge on personal health, it allows users to take an active part in maintaining their health and taking precautions as needed. As the device has features such as being portable and remotely accessibility, it makes devices like Smart Vitals to be effective and beneficial for any individual to keep track of their health, even on the areas with insufficient healthcare facilities.

Smart Vitals: A Portable Health Monitoring System, summarizes the importance of compact device, high-tech health monitoring system via advanced technology. It focuses on device's intelligence, accessibility with user friendly components making it advanced, accurate and efficient health measuring system made to achieve present days' requirements. The device helps to tackle obstacles related to healthcare management by allowing users to view their real time health vitals. One of the major benefits is the ability to minimize the need for medical visits for regular health check-ups, reducing expense and saving time. In todays' modern market, people are more inclined towards smart healthcare system including wearable technologies, Smart Vital distinguishes itself by presenting value for money, live tracking, and mobile application integration.

The "Android Based Health Parameter Monitoring" by (Trivedi & Cheeran , 2017) has embedded system with Android mobile app that tracks and keeps record of vital heath signs which includes body temperature and heart rate via sensors and transmits data via Bluetooth to mobile app. While Smart Vitals offers wide range of health parameters for monitoring including ECG, blood pressure and SpO₂ level. The data transmission is real-time based and quick in the mobile application via Wi-Fi connection. The mobile application developed by (Trivedi & Cheeran , 2017) is very minimal delivering limited user interaction, with basic display of readings. In comparison, Smart Vital offers more user engagement with visual analytics. The system has dashboard which allows users to view detailed data trends, elevating user experience allowing the app to be user-friendly and helpful.

According to (Varghese & Muthukumaraswamy, 2025)’s “An IoT-Based Health Monitoring System for Elderly Patients”, health care has been lacking behind a lot due to growth in elderly population in China, which is the reason for the development of remote health monitoring device. The device includes components for measuring pulse rate, blood pressure, temperature and oxygen saturation and Arduino as the core processing unit. Although the system by (Varghese & Muthukumaraswamy, 2025) offers most of what Smart Vitals offers, the system does not contain mobile application which is somewhat a necessary part as it allows users to view data anywhere with Wi-Fi connection. Smart Vital offers a feature to send alert message to the users for any irregularity in the reading, which allows users to take precautions to make sure they are maintaining their health status, which is lacking in the “An IoT-Based Health Monitoring System for Elderly Patients” research paper. The need for alert message is crucial as the system focuses on elderly patients who are more vulnerable to health issues.

“Development of Blood Oxygen Level, Heart Rate And Temperature Monitoring System by Using ESP32” system developed by (Ahmad, et al., 2022) has presented their system using ESP32 integrated with MAX30102 and MLX90614 for SpO₂, Beat Per Minute (BPM) and body temperature respectively, which will then be displayed on the android application. The ESP used in the development of the system is powered by using LIPO rechargeable battery via charger module. ESP 32 WROVER B used in the Smart Vital project has a built-in battery holder which can power the microcontroller without the need for extra components or modules. The results can be monitored on the Serial Monitor of Arduino IDE and via cloud system. One of the biggest drawbacks of the system I think is not using a remote tool for displaying data, additionally not including mobile application. Smart Vital project contains both OLED display and mobile application for user convenience. The accuracy rate of 95% for temperature, can be improved by using sensor such as per (Sari, et al., 2021), an accuracy of 99% was achieved with right calibration. Similarly,10K NTC thermistor thermometer sensor was used in Smart Vitals which gives more accurate reading when calibration is done right. (Ahmad, et al., 2022) has stated in their research paper on how the system can be further improved, by adding blood pressure measurements, which is implemented in Smart Vitals project.

(Sheikh , et al., 2024) has delivered a simple system in “Analysis of Patient Health Using Arduino and Monitoring System” for measuring vital signs such as temperature, heart rate and blood oxygen level, processing data via Arduino. While making any device which includes crucial data it is important for the developers to choose the components wisely specially health monitoring system. The study carried out by (Hosan, et al., 2025) concluded that ESP is reliable as it performs much better in comparison to Arduino while performing sophisticated tasks. “Analysis of Patient Health Using Arduino and Monitoring System” does not focus on development of mobile application, only the use of LCD is seen. Smart Vitals has mobile application which allows users to explore different features. (Sheikh , et al., 2024) has stated in their journal about inaccurate readings and less IoT features. It is essential to have nearly precise reading while developing such system so that users can have assurance and can take precautions accordingly.

# **CHAPTER 3: REVIEW OF TECHNOLOGY**

Different tools, technology and components were used in the project during the development of Smart Vitals which are listed below:

## **3.1 Hardware Components**

1. **ESP 32 WROVER TYPE B** is a micro-controller, it has powerful ability to process data with built-in Wi-Fi which was the reason for its usage.

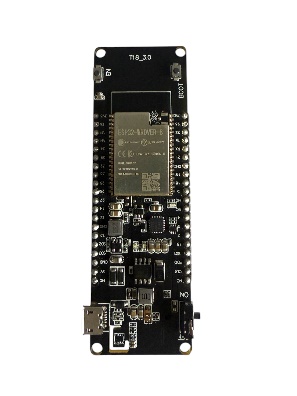


Figure 1: ESP 32 WROVER TYPE B

1. **MAX30102** **sensor** uses photoplethysmography (PPG) which was used to measure an estimated blood pressure.



Figure : MAX30102

1. **AD8232** is a device to measure ECG signal. It was used as it gives optimal ECG signal measurement.

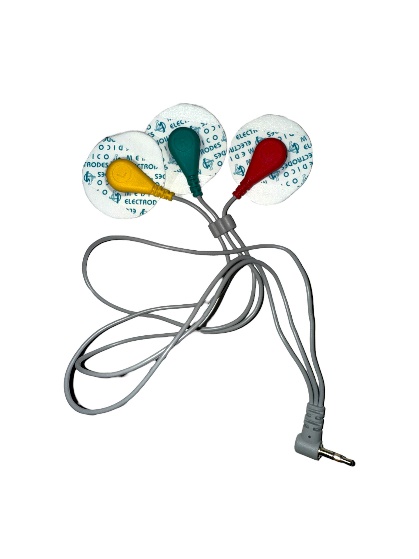
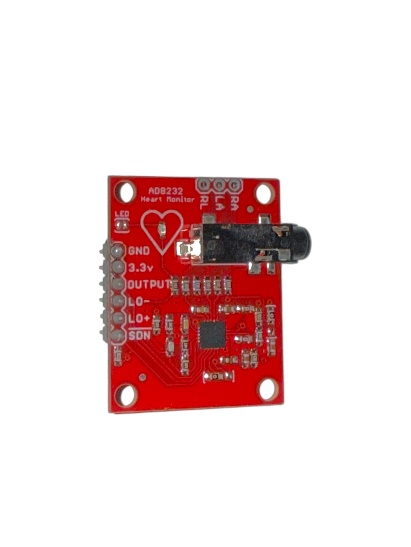


Figure : AD8232

1. **10K NTC Thermistor temperature sensor** is used for measuring body temperature. Its resistance lowers while temperature is increased, making it suitable for accurate temperature measurement.

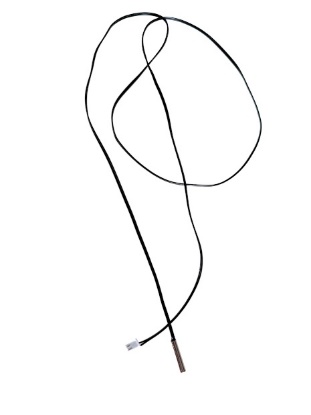


Figure : 10K NTC Thermistor temperature sensor

1. **Resistor** is electrical component which controls voltage and limit current. It ensured the components receive proper voltage and current supply.

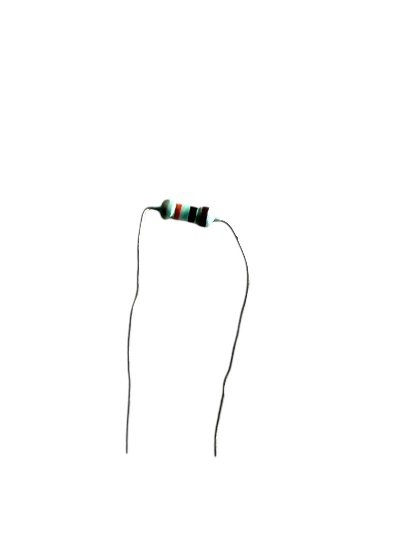


Figure : Resistor

1. **Push Button** is electromechanical switch, allowing manual control of the circuit. It was implemented to switch the sensors measurement as per requirement.



Figure : Push Button

1. **OLED** was used for local display of the real-time measurements.

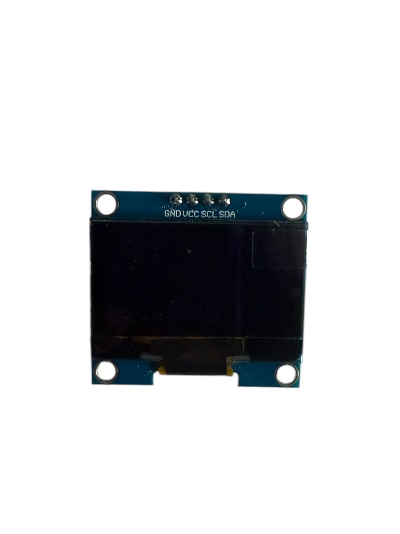


Figure : OLED

1. **Batteries** were used to provide a common power supply to make sure the ESP works properly with sensors.



Figure : Battery

1. **MQ-135** was used for air quality monitoring of a particular room, showing different ranges of air quality.

****

Figure : MQ-135

1. **NEO-6M GPS Sensor** was used as it allows users to track the live location, which is used as additional layer for safety of an individual.



Figure : NEO-6M GPS Sensor

1. **DHT22** was used to get knowledge on the room temperature.

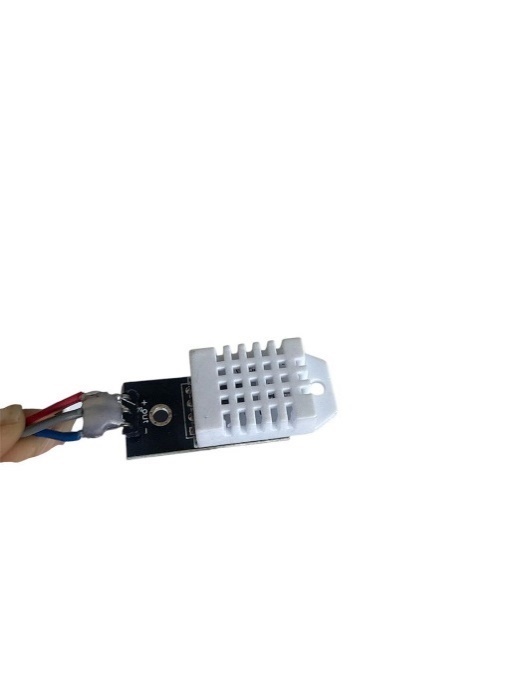
****

Figure : DHT22

1. **Matrix Board** was used to integrate all the components with ESP making the connection easier to join and compact.

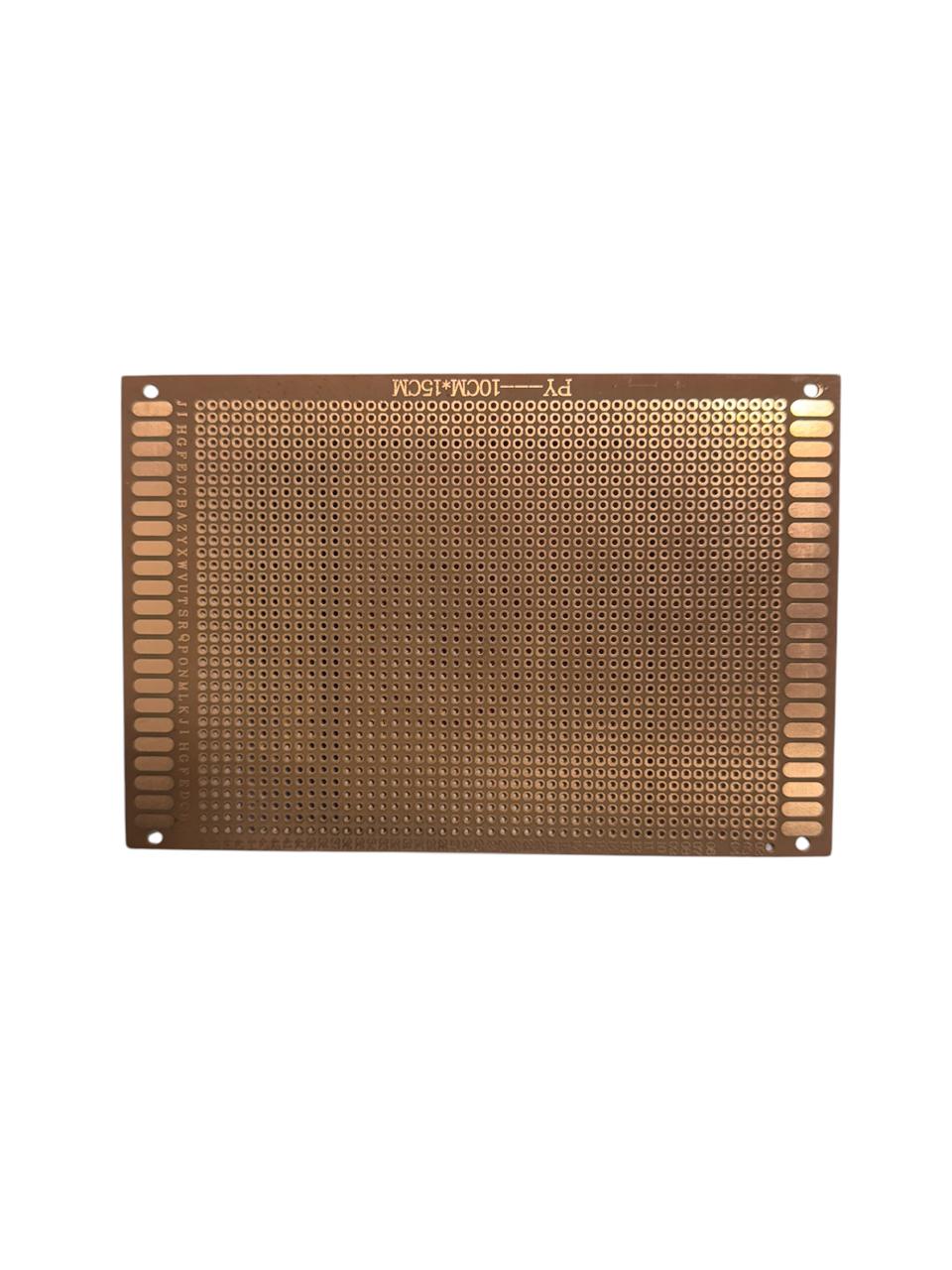
****

Figure : Matrix Board

1. **Soldering items** such as soldering rod and soldering wire, steel wire wool, wax was used to join the wire for fixed connections.

****

Figure : Soldering iron, wax, wire, steel wire wool

1. **Glue** was used to join all the components ensuring the connections to work properly and to attach the 3D printed box.

****

Figure : Glue Gun

1. **Connection Wire** was used alongside soldering to connect the ESP with hardware.

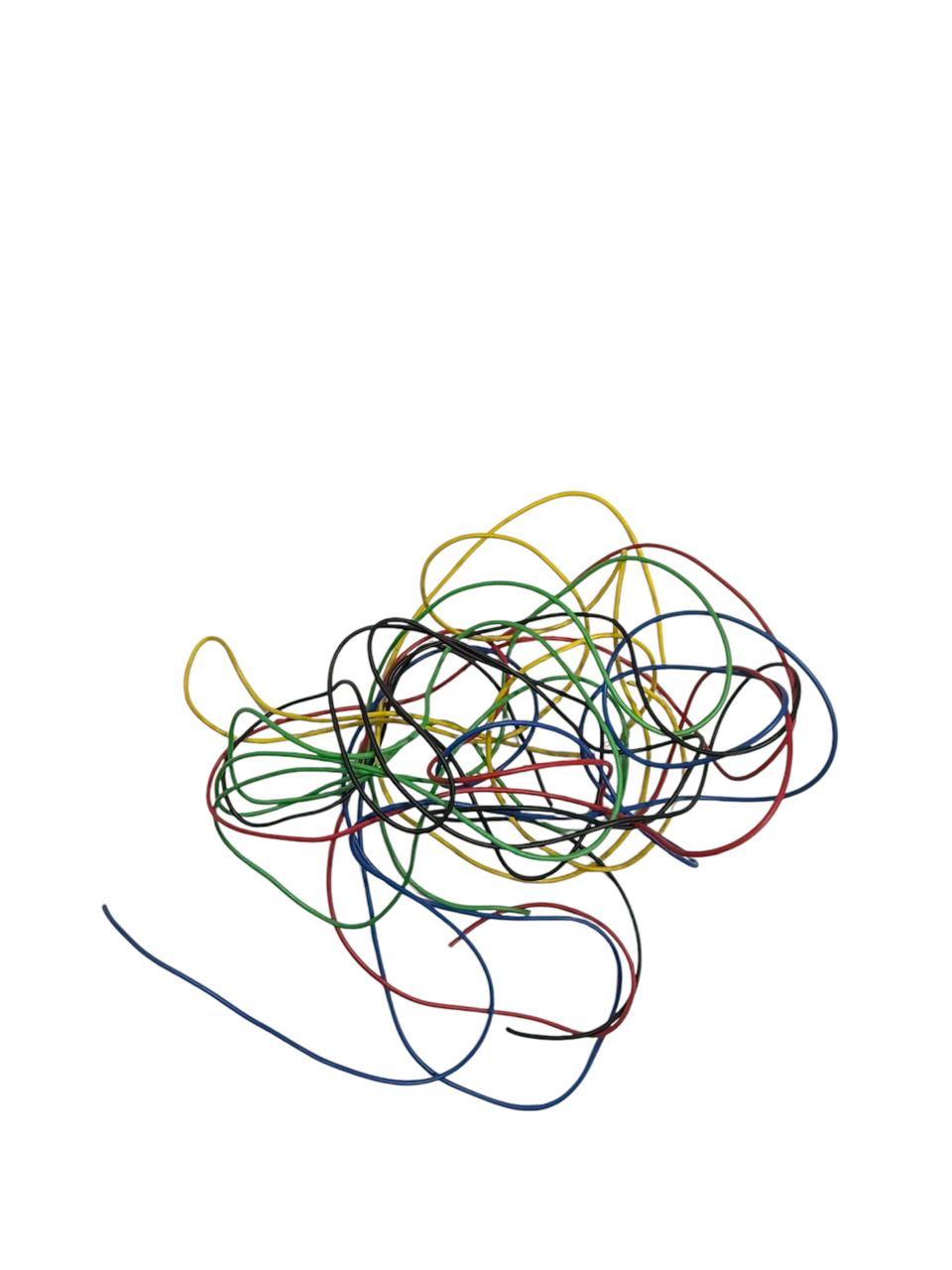
****

Figure : Connecting wire

1. **3D printing (Creality Ender 3)** was used to make a box which fitted all the components which made the project more presentable and easier to carry.

****

Figure : Creality Ender 3

1. **Power bank Module** was used to charge the battery via charging cable also power the system.

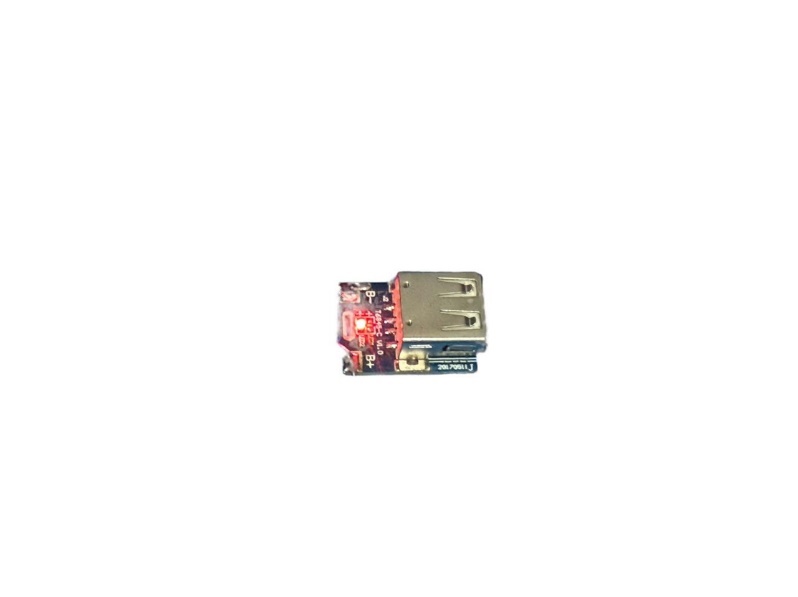
****

Figure : Power bank module

1. **Reset Button** was used as it allows to reset the ESP, which might come in handy when the system is lagging.

****

Figure : Reset Button

1. **Type B cable** was used to connect the system with charging cable which allowed the battery to be charged.

****

Figure : Type B cable

## **3.2 Backend**

1. **Firebase** was used as it acts as a secure platform for user authentication and stores users’ current health vital information ensuring smooth access to user data.

## **3.3 Operating System**

1. **Windows 11** is used as it supports all the software and hardware components for smooth operation.

## **3.4 Language, framework and Libraries**

1. **C/C++** was used as primary programing language as it helps in effective programing and suitable for hardware like ESP32.
2. **Dart** was adopted for development of mobile application.
3. **Flutter** was used for creating responsive UI for real-time data visualization.
4. **ThingSpeak** allowed to visualize, collect and analyze data after integration with microcontroller.
5. **ESP32-SDK** allowed effective development, handle wireless communication and hardware integration.
6. **Wire.h** was used for I2C communication for effective data exchange among ESP32 and components used.
7. **WiFi**.**h** library was used as it allows Wi-Fi connectivity for IoT devices which helps in data transmission.
8. **SparkFun** **MAX3010x** Library allowed easy communication with MAX sensor and collect accurate data in real-time.
9. **U8g2** library is common library for graphics display used with microcontrollers to get display in OLED.
10. **Math**.**h** library was used to carry out mathematical calculations which was essential during calibration and while using different formulas and equations.
11. **TinyGPS++.h** library was used to gather GPS data which helped in getting the location of the device.
12. **DHT.h** library was utilized while gather data from DHT22 sensor for temperature data.
13. **HTTPClient.h** library was used to receive and send data to and from the web server.
14. **HardwareSerial.**h library was used as it allows communication with various serial device using UART ports, which are built-in ports in ESP32.

## **3.5 Version Control**

1. **GitHub** was utilized as it helped to manage the codes and files efficiently and keep track of the project history and encouraged for regular commits.

## **3.6 Documentation**

1. **MS Word** was used to create initial plan, ethical consent form for the project and prepare a thorough report of the project.
2. **MS Project** was used to present a well-constructed workflow of the project for ease development of the project in a structured manner.
3. **MS PowerPoint** was used to deliver presentation which supports the development of the product.

## **3.7 Web Browser**

1. **Google Chrome** was utilized as it is easy to use, offers secure surfing experience and is trusted web browser for debugging and running web portals.

# **CHAPTER 4: METHODOLOGY**

Agile methodology has been used for Smart Vitals project. Agile method allows scalability and adaptability which enables to handle technical error effortlessly at early stages without interrupting other components. The flexibility, in Agile methodology allows developers to include new features or components in later stages based on the evolving requirements. Constant feedback is provided which ensures the progression of the project to meet the end goal the expectations. Additionally, the agile method offers iterative process enabling to build and improve the project piece by piece rather than following a linear or firm plan.

The Agile process has different stages each stage contributing for the progressive development of the Smart Vitals project. Research and Specification Collection is the first stage where the essential modules like ESP32, MAX30102, SpO₂ sensor, ECG sensor, Temperature sensor, are examined with the help of the datasheets. The next stage is Planning and Circuit Designing where the sensor connections were validated with the help of Wokwi for circuit diagram layout. After the successful circuit design, the main focus was integration of MAX30100, ECG and Temperature sensor with ESP32 for processing sensor data and with OLED for data monitoring. Then, 3D models and PCB design was developed and designed in TinkerCAD during Designing and Prototyping stage. In the Development stage, the microcontroller was programmed to analyze sensor data to communicate wirelessly with the mobile app. After the system being functional, in the Testing phase the product was regularly tested ensuring precision, stability and working of hardware and software components. Lastly, in deployment and maintenance stage, improvement of the product was done where the product was updated based on user feedback.

The waterfall method is a conventional and sequential approach where each stage must conclude before beginning with the next stage following stable workflow used for project that has well-defined end goal. The project Smart Vitals project has different hardware and software integration which needs to be tested continuously for precision and improvement, in waterfall method testing is conducted following the development stage. Any issue in the project might not be identified until the very end which will be very risky making the process time consuming and complex which is the reason for Agile methodology being used as it offers constant testing and evaluation throughout the product development. RAD focuses on prompt development of the prototypes and user reviews. The Smart Vitals project focuses on providing accurate, dependable and secure data which are essential for health care, RAD focuses on quick deployment which could result in patchy testing and disparity between hardware and software. Therefore, Agile methodology is used as it provides iterative feedback which assures the product is tested thoroughly and reliable medically. The Spiral Model mainly focuses on risk management and developing the project in an iterative manner which can help the developers eliminate or avoid the risks completely during the initial phase. It focuses on thorough reports, planning which is suitable for large-scale project with peak risks which is why agile method is used as it is suitable for small to medium scale projects like Smart Vital with risks like data reliability which can be avoided with regular evaluation.

# **CHAPTER 5: PRODUCT DESIGN**

## **5.1 General Overview of the system**



Figure : Smart Vitals device

Smart Vitals system starts the work when the device is turn on via switch, which activates the micro controller then the OLED display. The OLED asks the user to press the button according to their preference of vital measurement which then activates the sensor, the sensors start taking the reading. The reading via sensor is seen on the OLED display. The user can press the button again to deactivate the reading. Additionally, reset button has been placed which allows the users to reset the system when the system does not response. For charging the system, Type B cable can be used.

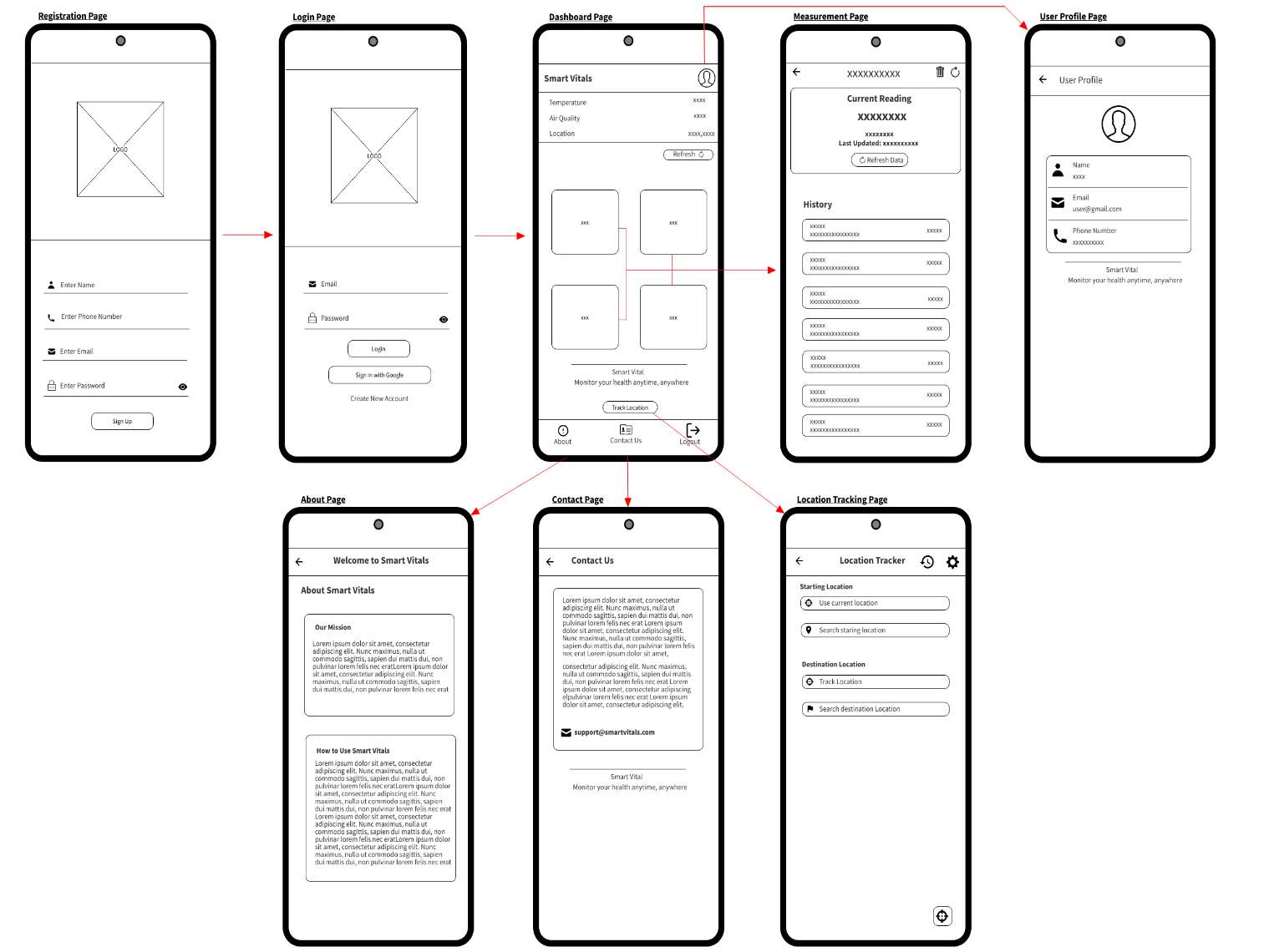


Figure : Wireframe for Smart Vitals

When the user logins on the Smart Vital mobile application, they can get access to their desired reading. If the user is not already registered the user can create a new account with valid credentials to get access to their vital readings. Additionally, the user can also view their vital history. The user can also get access to their profile, get general overview of the Smart Vital, how to use it on about page. The contact us page is also seen on the interface which allows user to get in touch if any queries. The user gets alert like voice alert and mail alerts if there are any abnormalities in the reading.

## **5.2 Data Flow Diagram**

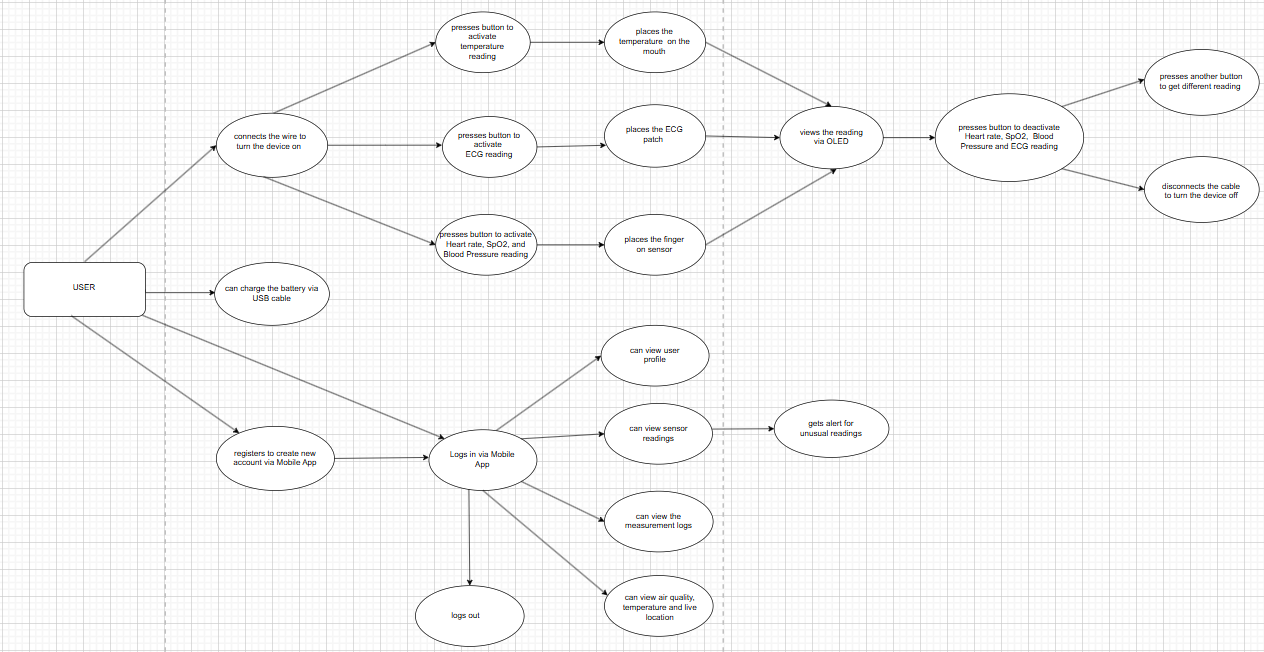
****

Figure : Dataflow diagram for Smart Vitals

Smart Vitals lets users to view their health vitals remotely and via mobile application which allows them to stay on top of their health and take necessary precautions. The smart vital is designed in such a way that the users can easily operate the device. To take any measurement the user first needs to turn on the device via button, which is present on the side. Once the device is turned the user can see the OLED being turned on, asking them for the specific vital they want to measure. The user needs to press the button according to their need of measurement, SpO2, Heart rate, Blood Pressure or ECG or body temperature. The system starts taking the reading once user has placed their finger on the sensor for SpO2, heart rate and Blood Pressure reading, or placed the patches on for ECG reading or placed the thermometer for temperature reading. The user can view their health readings on the OLED for remote display. Once the reading is done, they can either deactivate the sensor by pressing the button or turn the device off. The user can also charge the battery via USB cable.

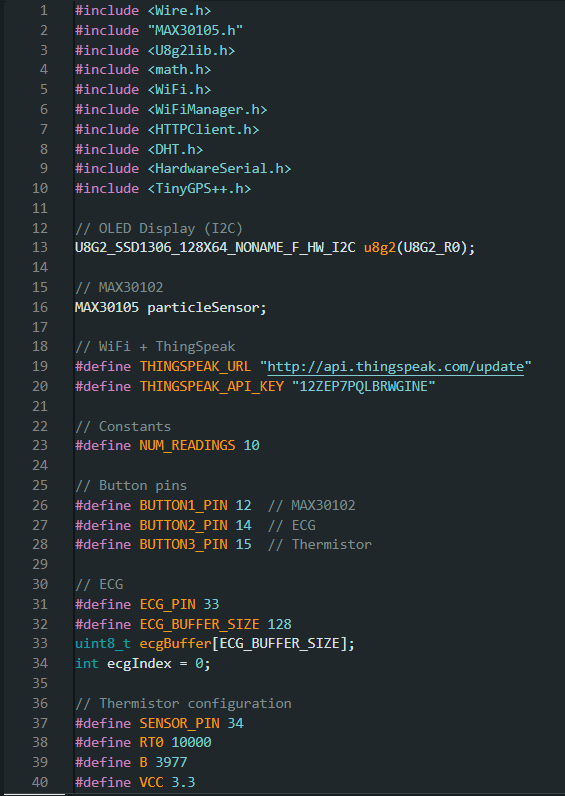
Additionally, the user can also register to the mobile application of smart vitals with their credentials or login if they already have an account with their correct credentials, which will redirect them to the smart vitals’ application. The user can view their health vitals via mobile application, all the measurements are separated so that it is easier for user to view the desired readings. They can also view their profile, can check the health measurement history. User can also view about smart vitals page to get information related to device. The user can get insights on the air quality, environment temperature and live location. The user can log out of the system with ease at any moment.

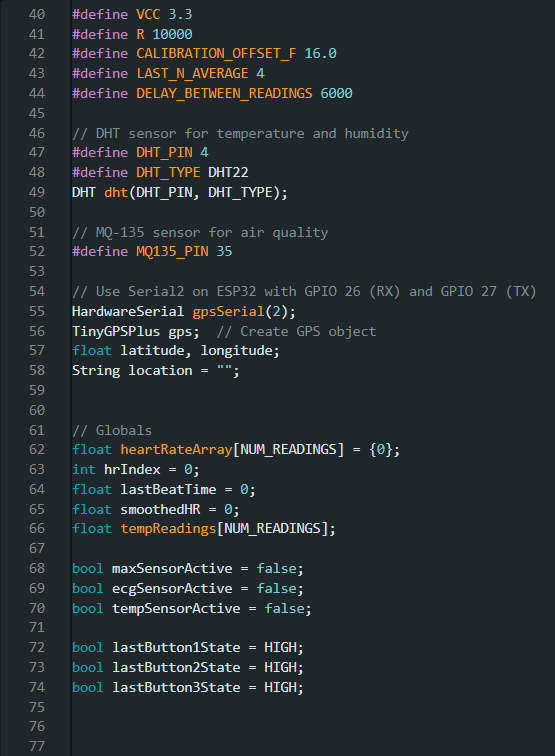
# **CHAPTER 6: SOFTWARE REQUIREMENT ANALYSIS**

The table below represents the features of the project with its priority listed as HIGH, MEDUIM OR LOW. HIGH being the most important feature, followed by MEDIUM being moderately important and LOW being relatively less important feature of the product.

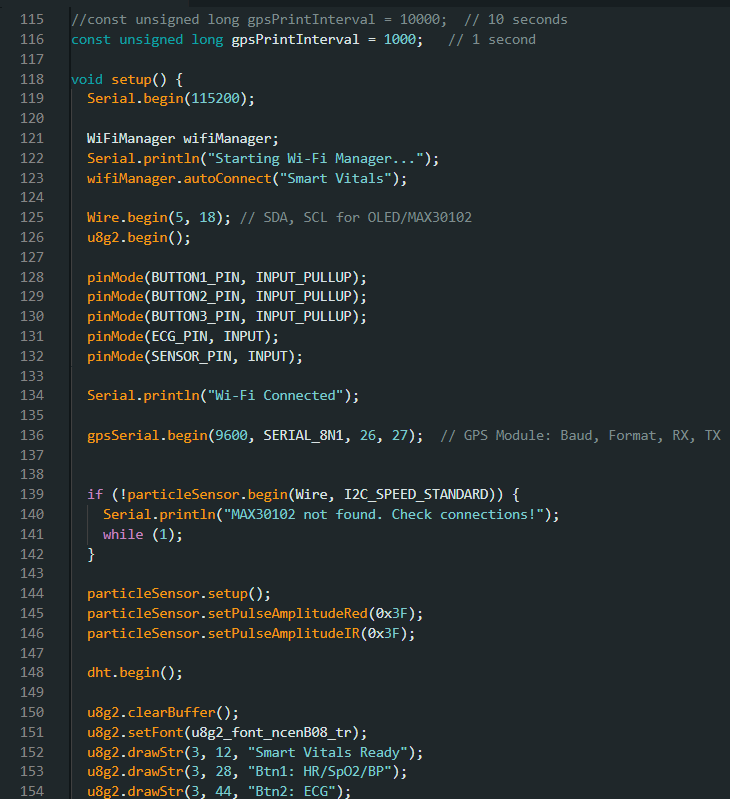
|  |  |
| --- | --- |
| **Features** | **Priority** |
| The device should be easy to use and portable. | HIGH |
| Cable for turning the device on and off. | HIGH |
| The device should connect to Wi-Fi for flawless communication. | HIGH |
| Buttons for switching between the sensors for particular health measurement. | HIGH |
| Readings from sensors (MAX30100, 10K NTC Thermistor Thermometer sensor, AD8232). | HIGH |
| The reading taken via sensor should be displayed on the OLED. | HIGH |
| The measurements should be shown in the mobile application through Wi-Fi connection. | HIGH |
| Alert should be sent to the user if there is any abnormality in the reading. | MEDIUM |
| The data from the measurements should be logged. | MEDIUM |
| The device should be able to charge when connected using USB cable Type B. | MEDIUM |
| Users can get access to the live location from mobile application when device is turned on and connected to a stable Wi-Fi connection. | MEDIUM |
| Additionally, user can view location history. | LOW |
| Users can get insights on room temperature. | LOW |
| Air quality monitoring. | LOW |

# **CHAPTER 7: IMPLEMENTATION AND TESTING**

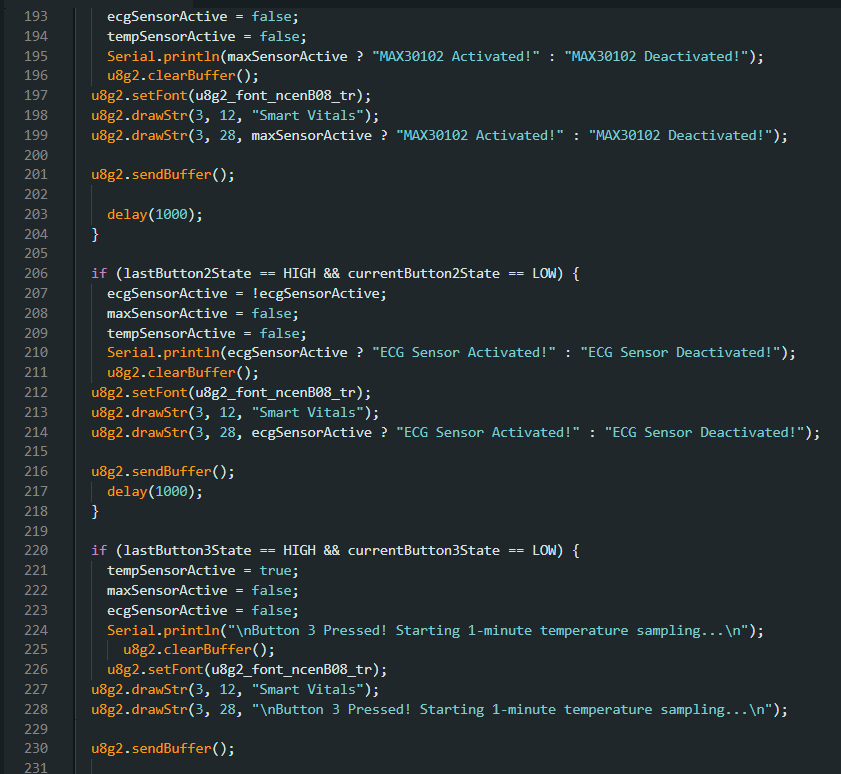






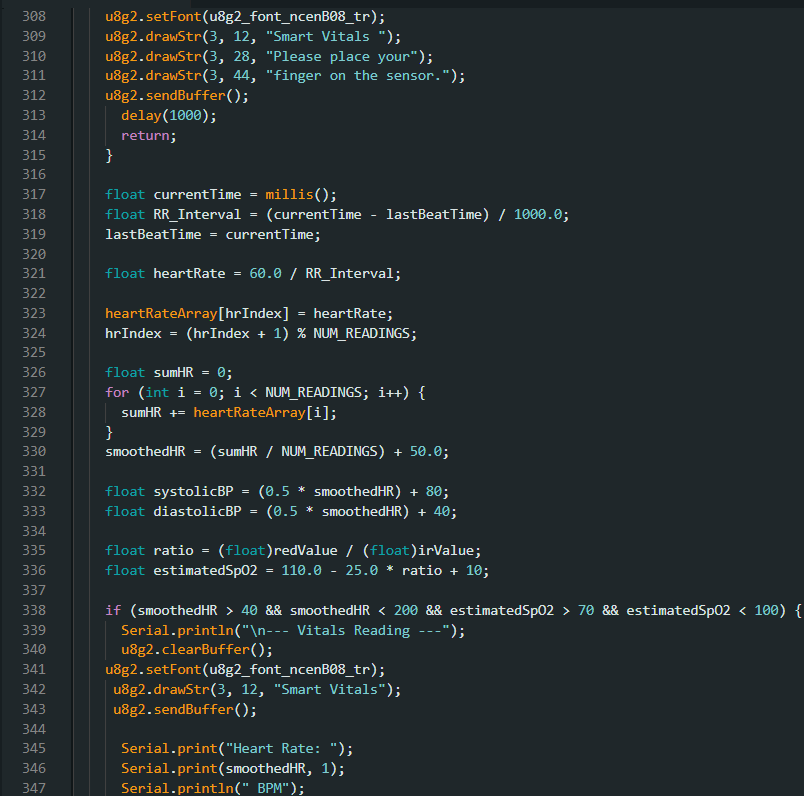
****

****

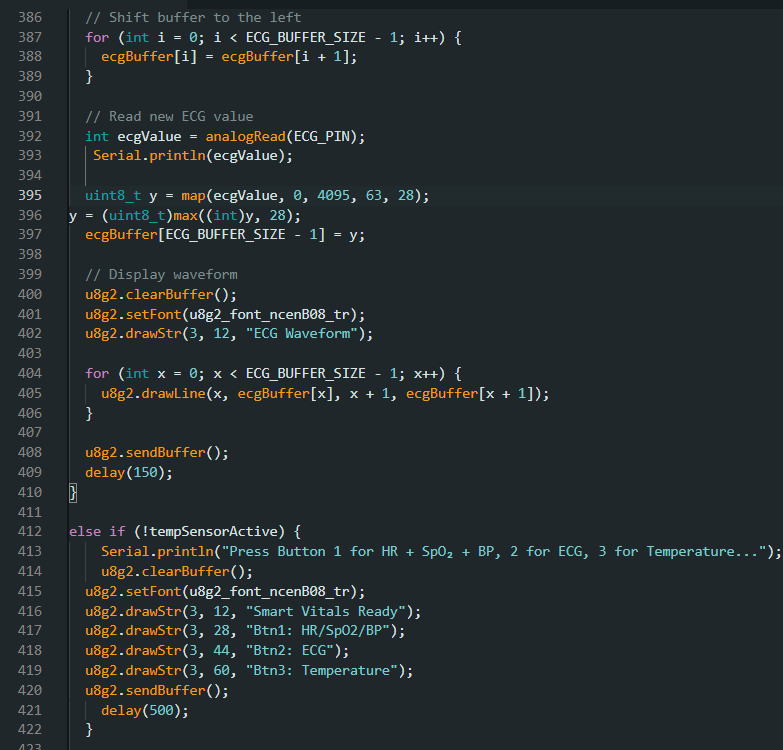
****

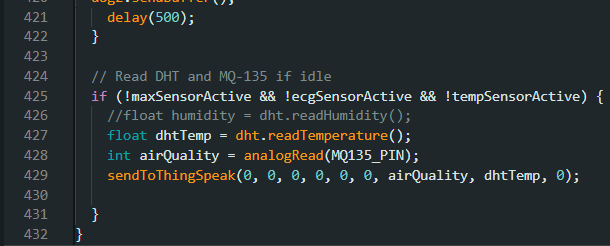
****

****

****

****

****

****

The above screenshots are the code for the hardware components, all the working mechanisms has been arranged in a proper systematic way i.e. all the inclusion of libraries has been declared in the beginning followed by configuration parts, and then begin the working mechanism of the sensors.

# **CHAPTER 8: PRODUCT EVALUATION**

In this chapter, features of the system have been listed for device and for mobile application separately with their expected results, the outcome of those features and evaluation to know whether the feature is performing as stated or not.

## **8.1 For Device**

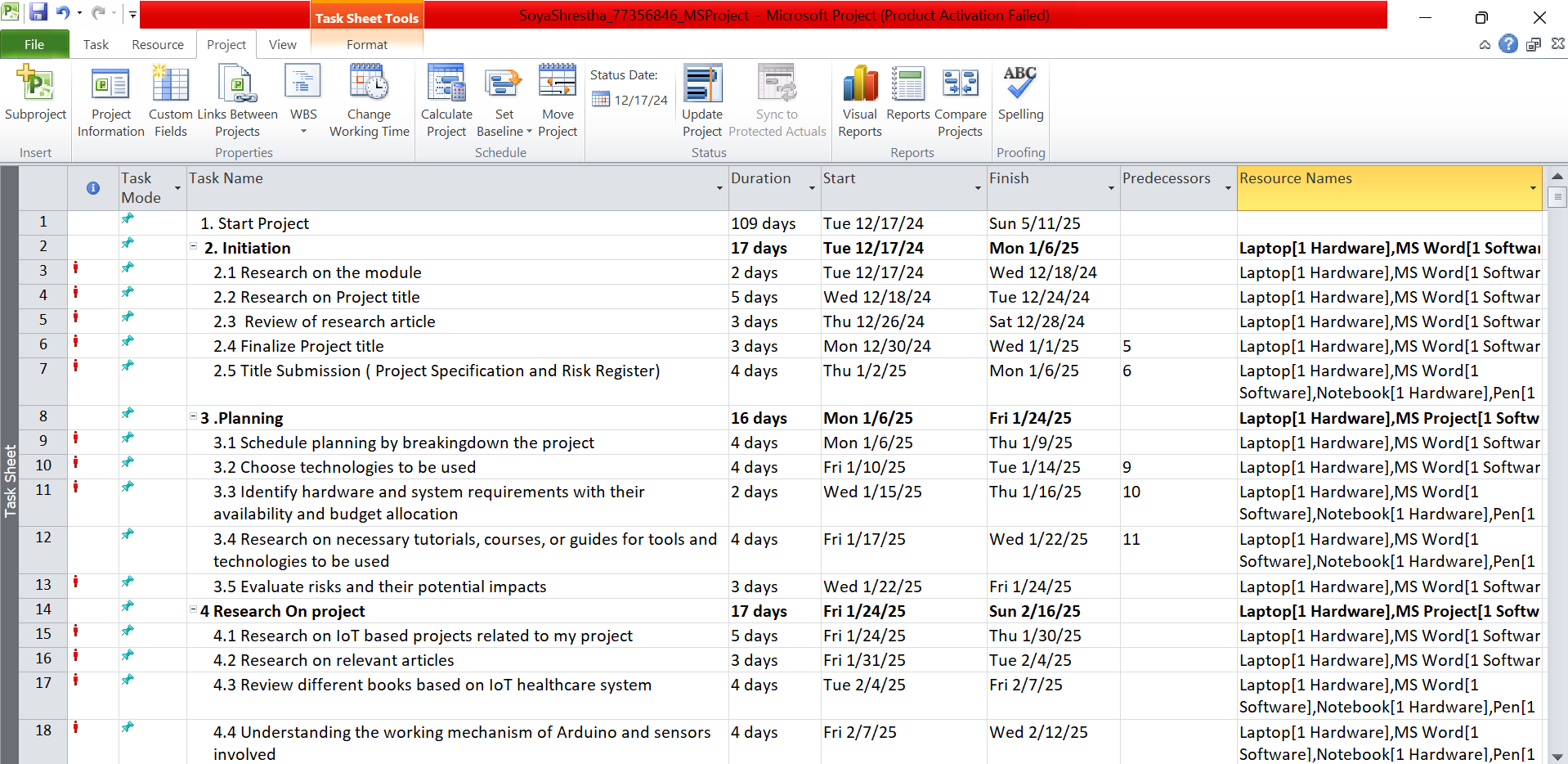
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Features** | **Expected** | **Outcome** | **Evaluation** |
| 1 | Connect the cable | The device should turn on and OLED start the display. | The device and OLED is turned on. | **Pass** |
| 2 | Press the button for heart rate, SpO2 and Blood Pressure | OLED should show MAX30102 is activated. | OLED displayed “MAX Sensor is activated”. | **Pass** |
| 3 | Place the finger on the sensor | The device should start taking the reading and display it on the OLED screen. | OLED started displaying the readings. | **Pass** |
| 4 | Remove the finger from the sensor | Should Ask user to place the finger on the sensor. | “Please place your finger on the sensor” message is displayed. | **Pass** |
| 5 | Press the button for heart rate, SpO2 and Blood Pressure again | OLED should show MAX30102 is deactivated. | OLED displayed “MAX Sensor is deactivated”. | **Pass** |
| 6 | Press the button for Temperature | The device should start taking measurement for body temperature, the reading should be visible via OLED. | The device stated taking the temperature measurements and displayed on the OLED. | **Pass** |
| 7 | Press the button for ECG | The device should activate ECG sensor. | OLED displayed “ECG is activated”. | **Pass** |
| 8 | Place the patches | The ECG waveform should be visible via OLED. | OLED displayed ECG waveforms. | **Pass** |
| 9 | Press the button for ECG again | The device should deactivate ECG sensor. | OLED displayed “ECG is deactivated”. | **Pass** |
| 10 | Connect Type B cable | It should charge the battery that powers the device. | Battery is charged. | **Pass** |
| 11 | Press the reset button | It should restart the system. | System is restarted. | **Pass** |
| 12 | Disconnect the cable | The device should be turned off. | The device turned off. | **Pass** |

## **8.2 For Mobile Application**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Features** | **Expected** | **Outcome** | **Evaluation** |
| 1 | Register | User should be able to create a new account. | New account was created. | **Pass** |
| 2 | Login | The app should be redirected to dashboard page when logged in with correct credentials. | Logged in via valid user credentials. | **Pass** |
| 3 | Click on SpO2 reading card | Should be redirected to different page to view SpO2 reading. | SpO2 reading is seen. | **Pass** |
| 4 | Click on Blood Pressure card | Should be redirected to different page to view blood pressure. | Blood pressure reading is seen. | **Pass** |
| 5 | Click on Temperature card | Should be redirected to different page to view temperature reading. | Temperature measurement is seen. | **Pass** |
| 6 | Click on Pulse rate card | Should be redirected to different page to view pulse rate. | Heart rate reading is seen. | **Pass** |
| 7 | Click on User icon | The user should be redirected to user profile page. | Redirected to user profile page. | **Pass** |
| 8 | Click on about icon | The user should be redirected to about us page. | Redirected to about us page. | **Pass** |
| 9 | Click on contact us icon | The user should be redirected to contact us page. | Redirected to contact us page. | **Pass** |
| 10 | Click on logout icon | The app should navigate back to login screen. | Redirected to login page. | **Pass** |
| 11 | Click on Track Location | The user is navigated to device location tracking page. | Redirected to device tracking page. | **Pass** |

# **CHAPTER 9: PROJECT EVALUATION**

## **9.1 Task Sheet**

****

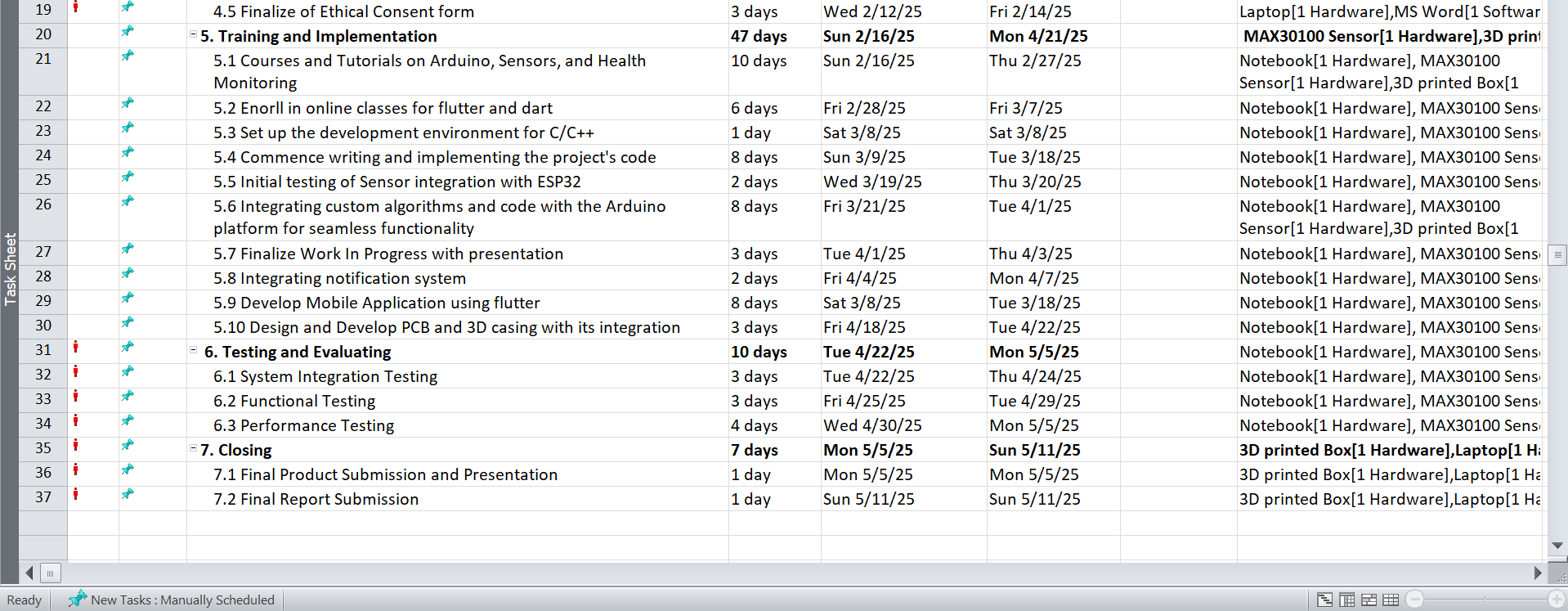
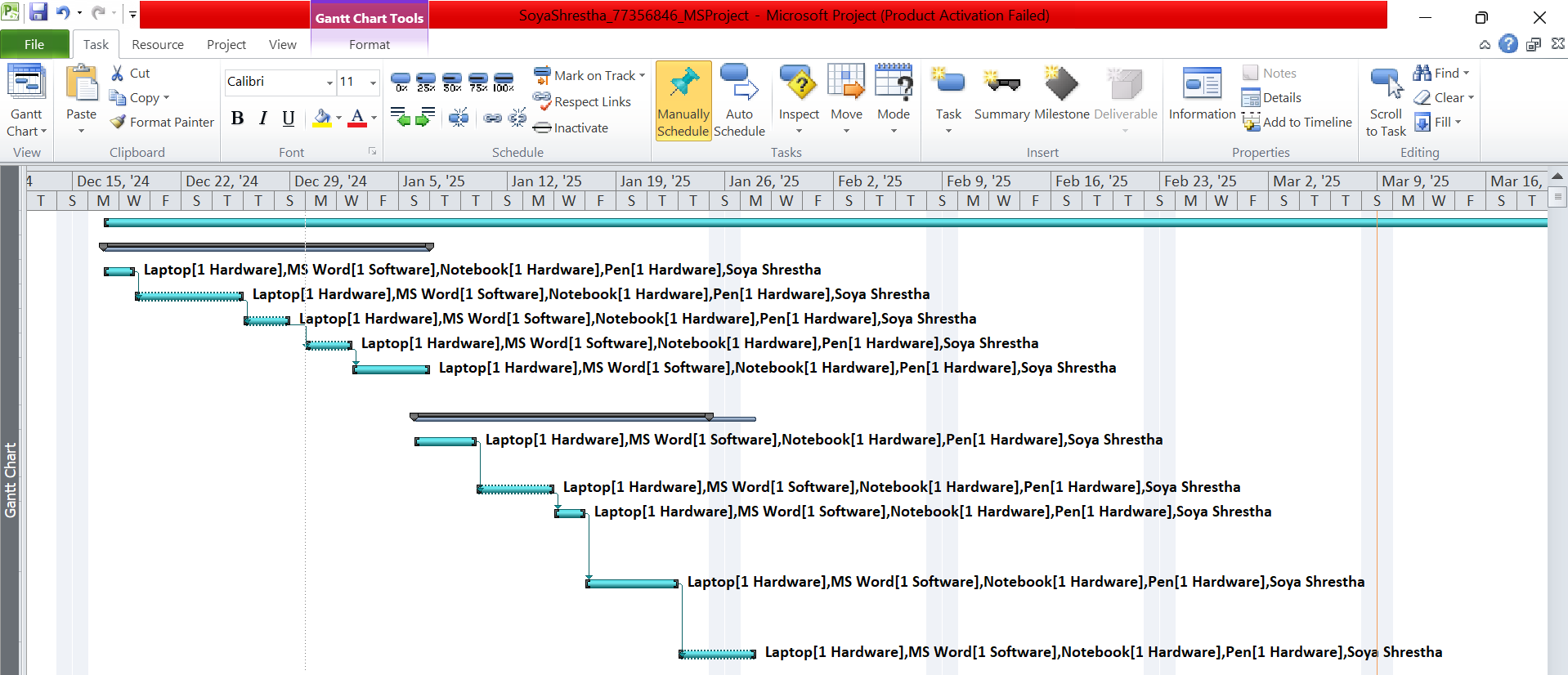
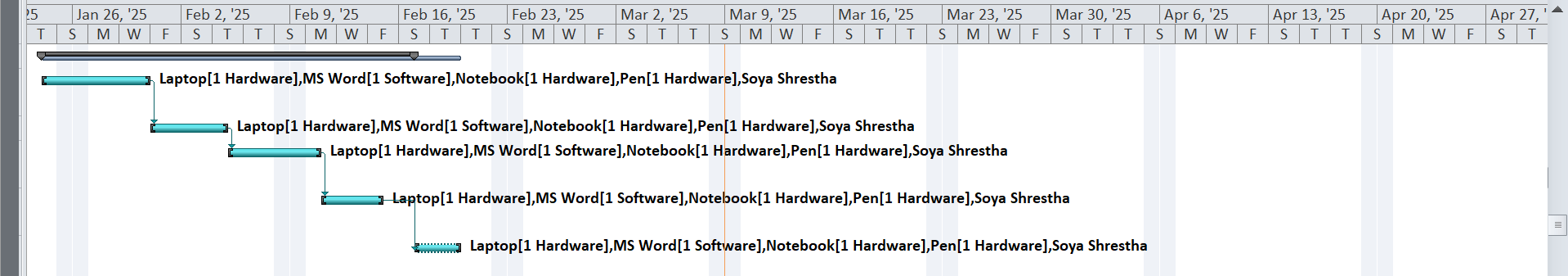
****

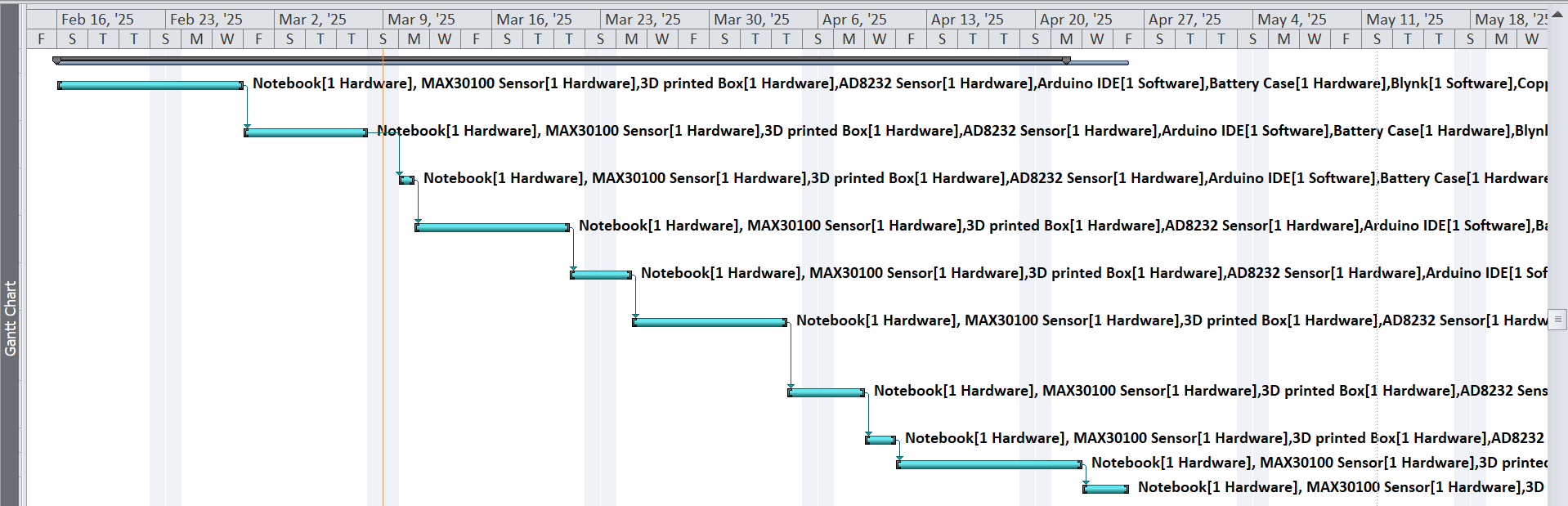
Figure : Task Sheet

The above figure represents the complete flow of how the project was successfully concluded by following a planned task sheet.

## **9.2 Gnatt Chart**

****

****

****

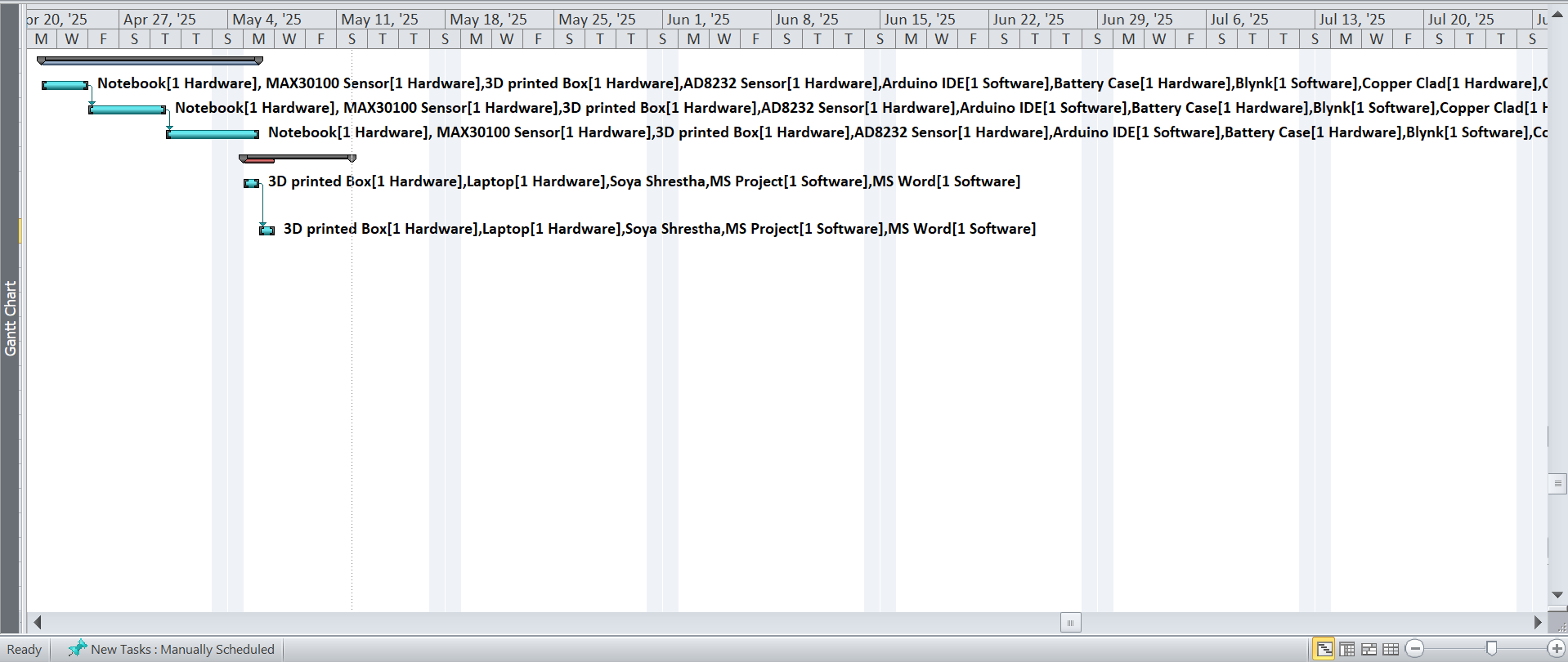
****

Figure : Gnatt Chart

The Gnatt Chart from the above figure is a visual display of the project flow, which tracks the tasks from start to the end.

## **9.3 Timeline**

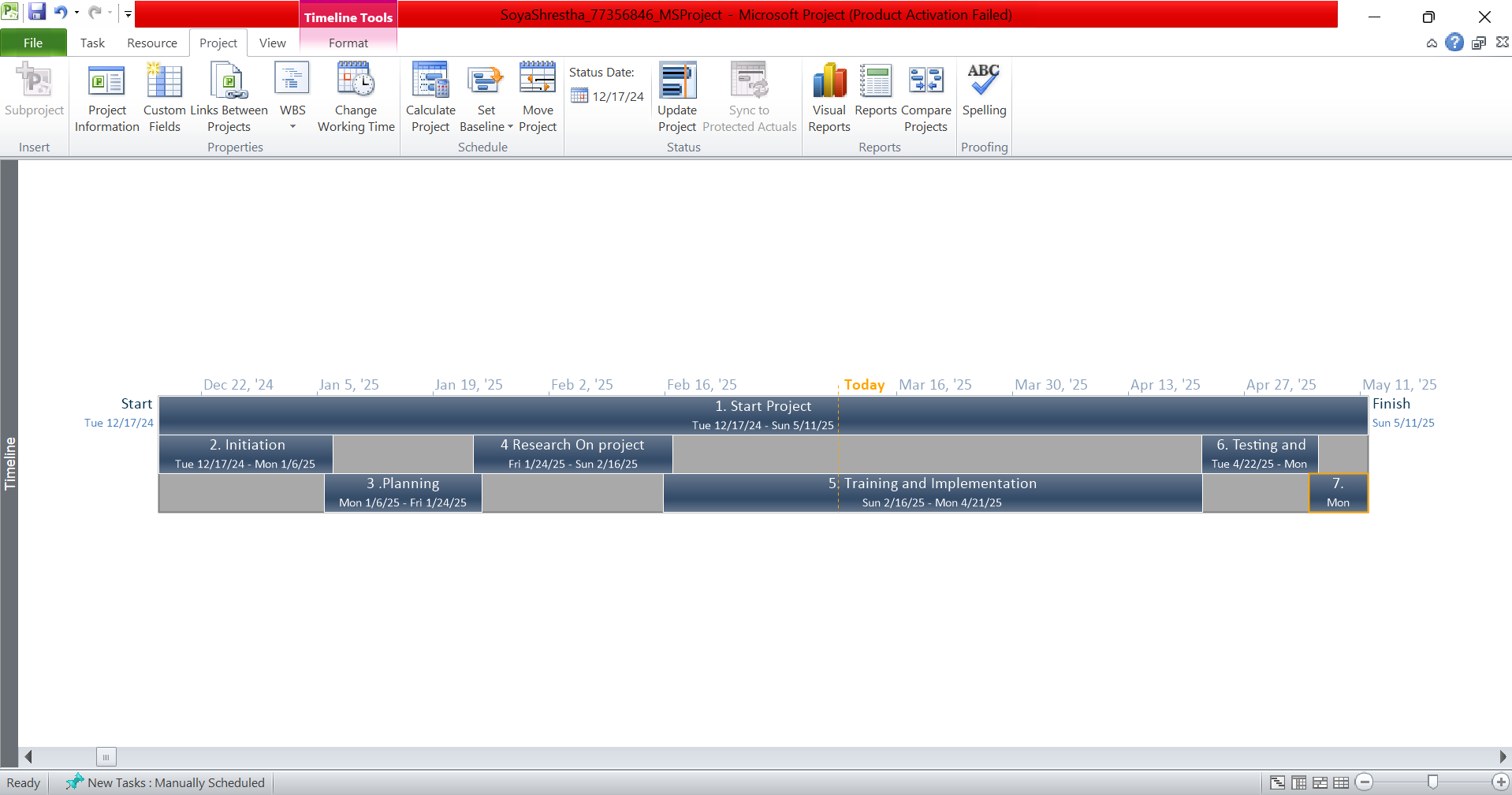
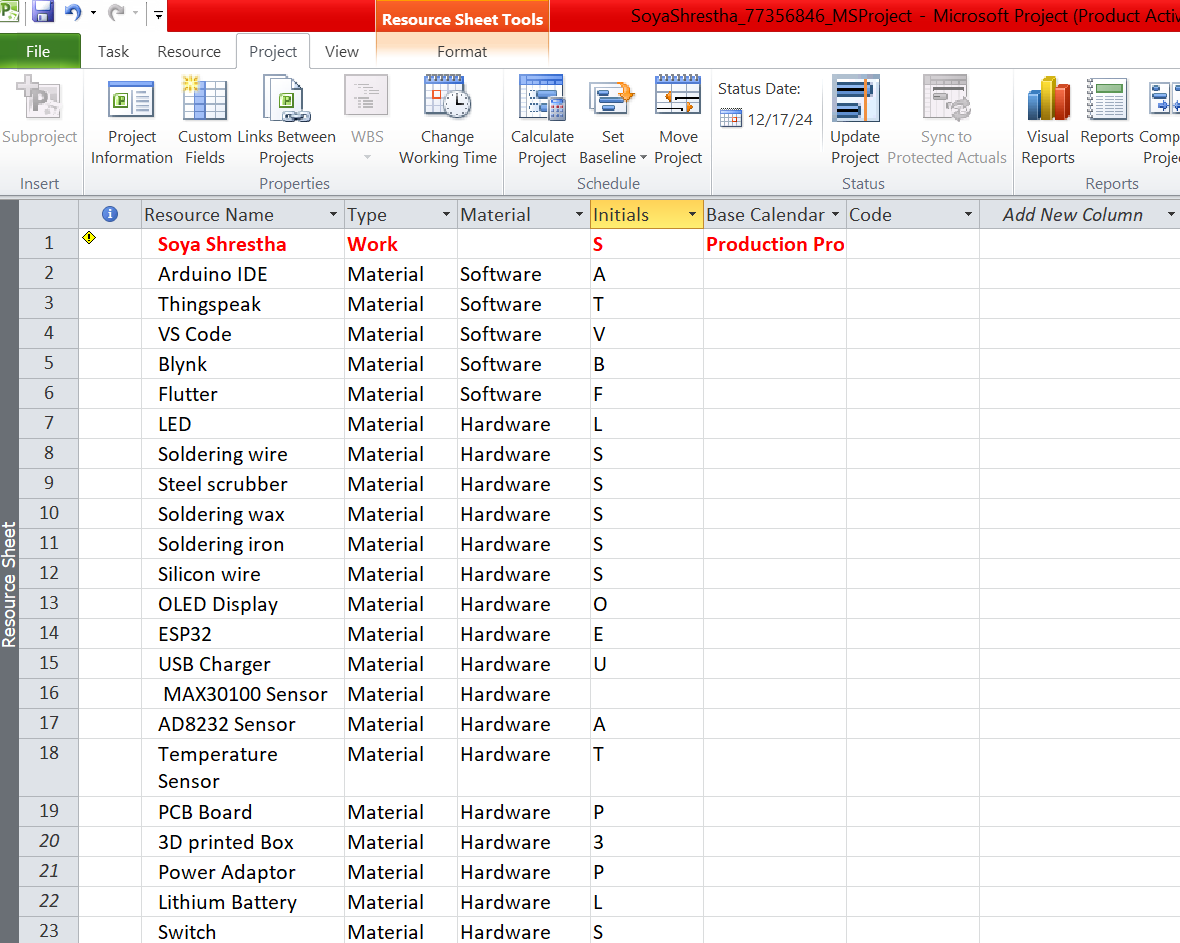
****

Figure : Project Timeline

Key stages of the project are shown in the MS Project Timeline, with initiation phase starting at 12/17/2024, followed by Planning phase, Research phase, Training and implementation phase, Testing and Evaluation phase with Closing phase marking the end of the project at 05/11/2025.

## **9.4 Resource Sheet**

****

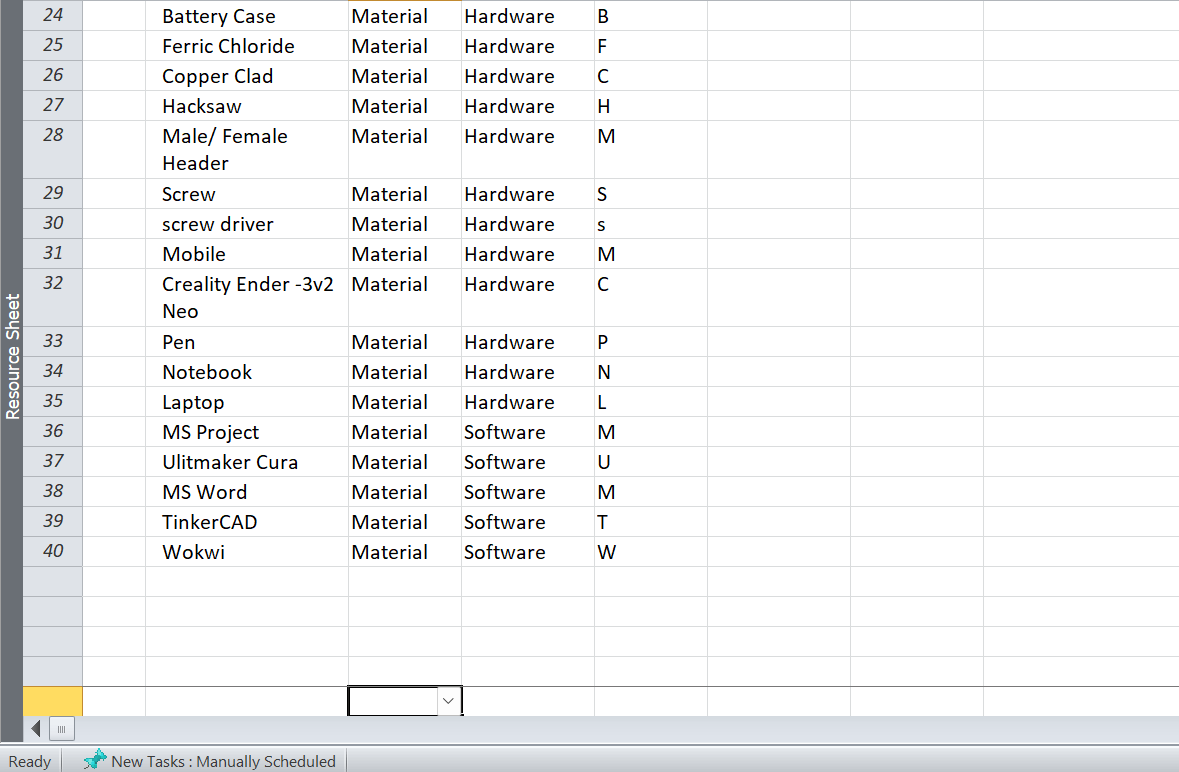
****

Figure : Resource Sheet

The above figure consists of all the tools and technologies (Hardware and Software) used, which are essential to complete the project.

## **9.5 GitHub**

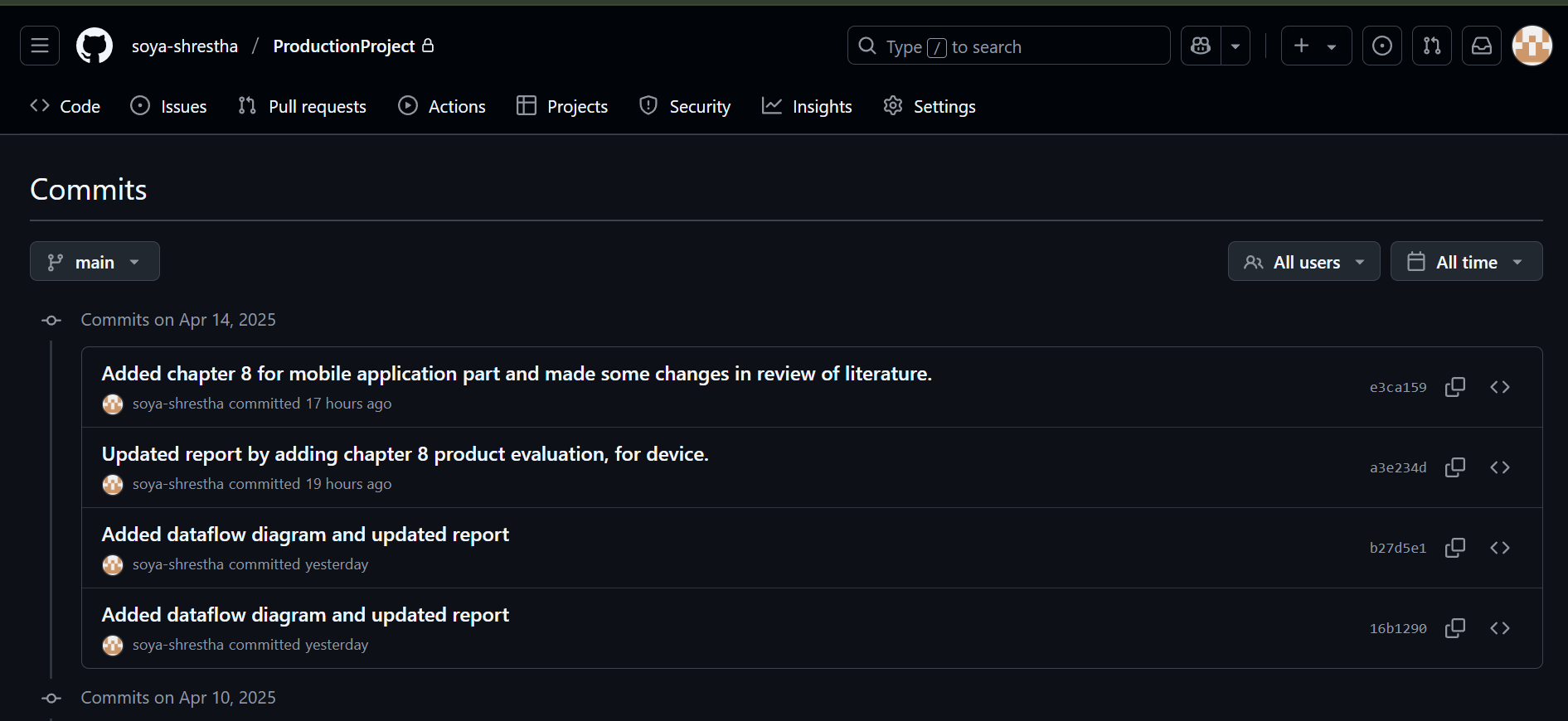


Figure : GitHub

GitHub was used to manage the codes and files in a systematic way ensuring regular commits to see the progression of the project.

## **9.6 Entity Relationship Diagram (ERD) and Composite ERD**

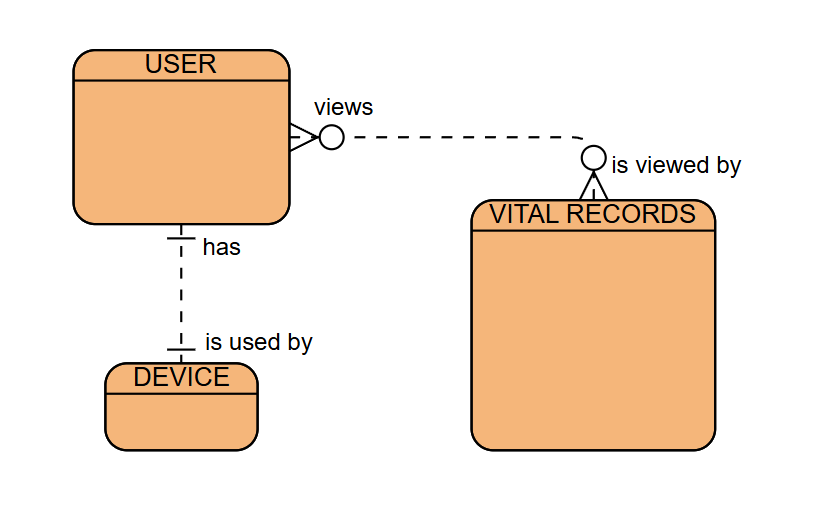


Figure : Entity Relationship Diagram (ERD)

The above ERD demonstrates the basic structure of how user is associated with the device and their vital records. A device is used by a user to get their vital readings. The generated vital records can be viewed by the users.

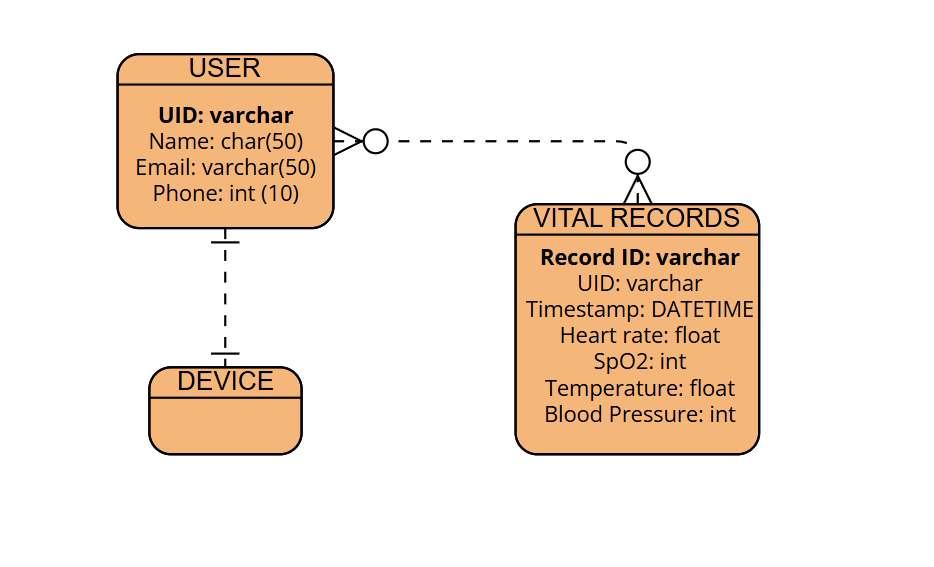


Figure : Composite ERD

The above figure is the Composite ERD of Smart Vitals, which shows a detailed relation between entity and their attributes; each user has their own unique UID with personal data. Vital records of each user are linked via UID (primary key) which is the foreign key to the vital records similarly Record ID is the primary key to the Vital Records entity. The entity “device” is linked with the entity “User” with one – one relationship (1:1). Similarly, “User” entity is linked with entity “Vital Records” with many – many relationship (M: N).

**9.7 UML**

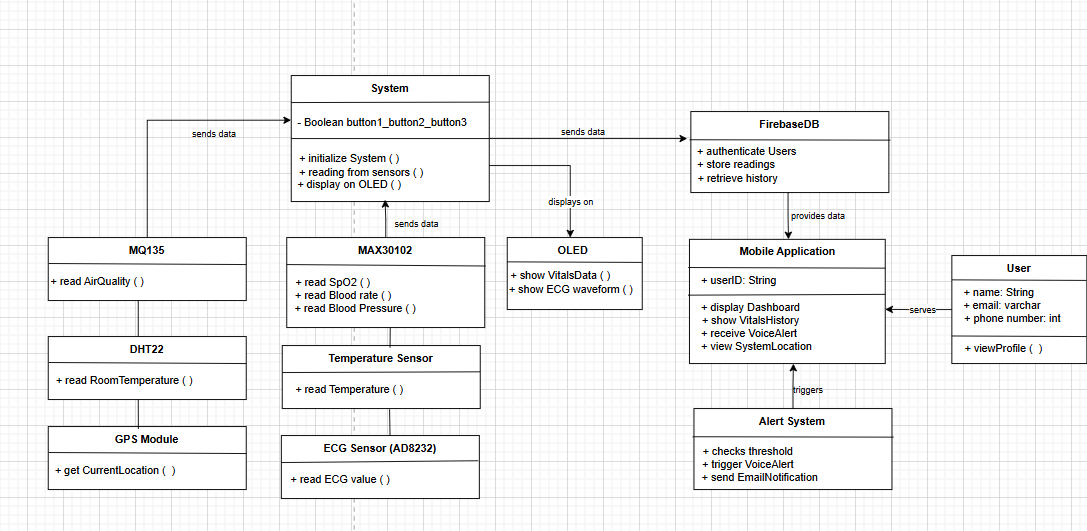


Figure : UML for Smart Vitals

The UML diagram below displays the structure of the Smart Vitals. The system module communicated with multiple sensors to measure health vitals in order to retrieve data via button press and environment conditions. The data retrieved via sensor are displayed on the OLED, concurrently sending data to Firebase database. The Firebase DB is then used to display data and history via mobile application linked to particular user. Additionally, alert via voice and email notification is triggered when reading is below threshold.

## **9.8 Requirement Catalogue**

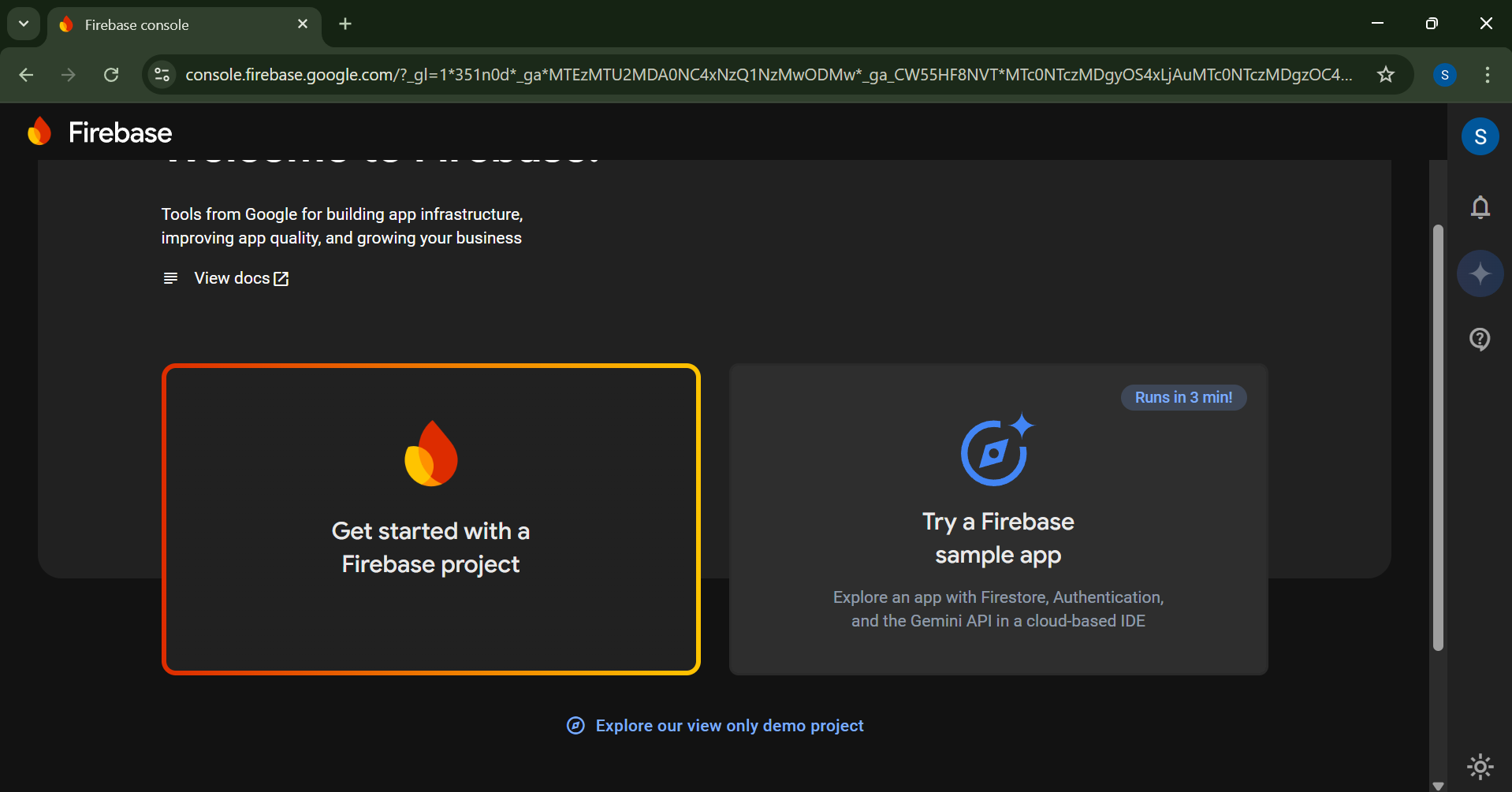
|  |  |
| --- | --- |
| **Functional Requirement** | **MoSCoW** |
| For flawless interaction the device must be linked with the Wi-Fi. | **M** |
| Buttons to start the measurement of vitals through various sensors. | **M** |
| OLED display to get easy and quick local access to the real-time health vitals. | **M** |
| Mobile application should be able to display vital readings. | **S** |
| Voice alert via mobile application for any irregularities in the readings. | **S** |
| Alert via email for unusual readings. | **S** |
| The mobile application could be able to get knowledge on air quality and room temperature. | **C** |
| Users could get insights on the location of the device. | **C** |

|  |  |
| --- | --- |
| **Non-Functional Requirement** | **MoSCoW** |
| Smart Vitals must be convenient, compact and easy-to-use. | **M** |
| The OLED must display the measurements on the spot. | **M** |
| The battery operating the Smart Vitals should last for a day or 2. | **S** |
| Using the type B cable the battery should be charged. | **S** |
| Further additional of the sensor could be supported by the device. | **C** |

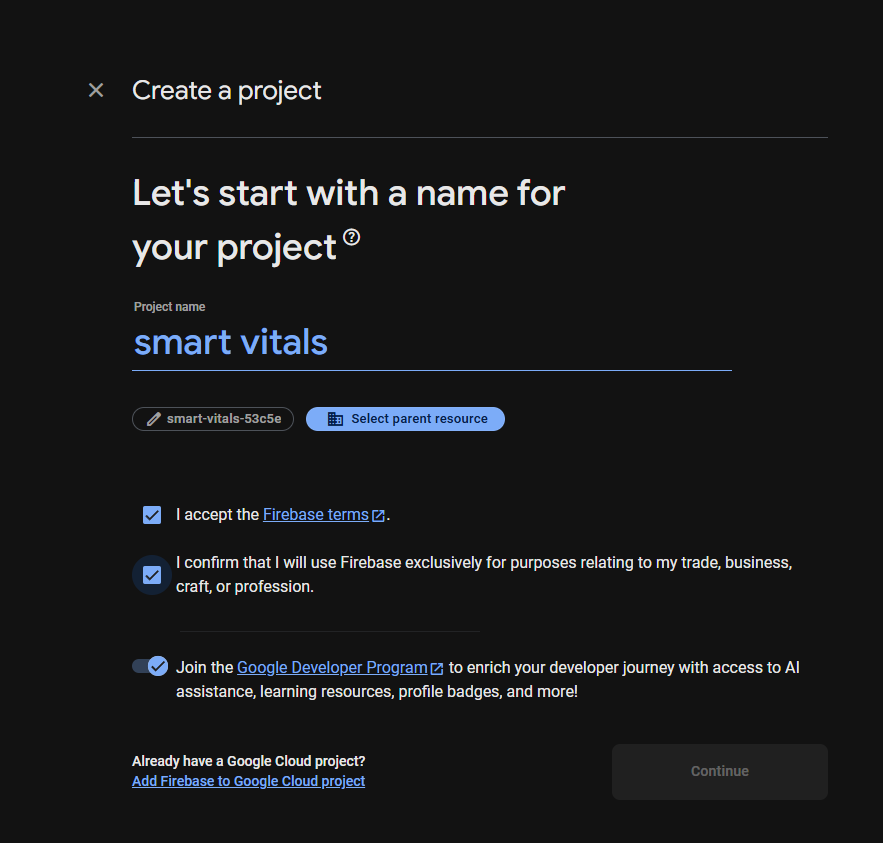
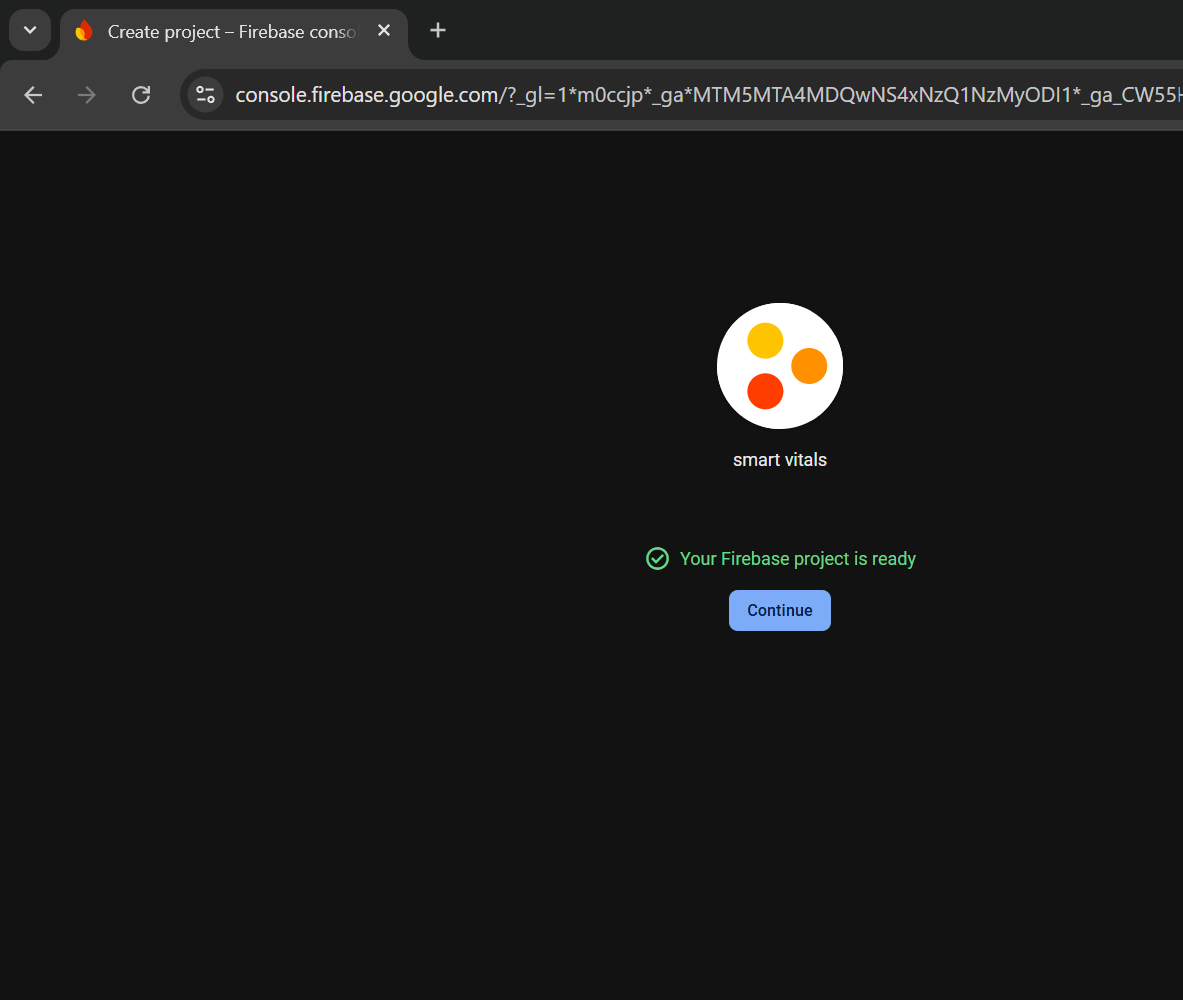
## **9.9 Installation Guidelines**

### **9.9.1 Firebase**

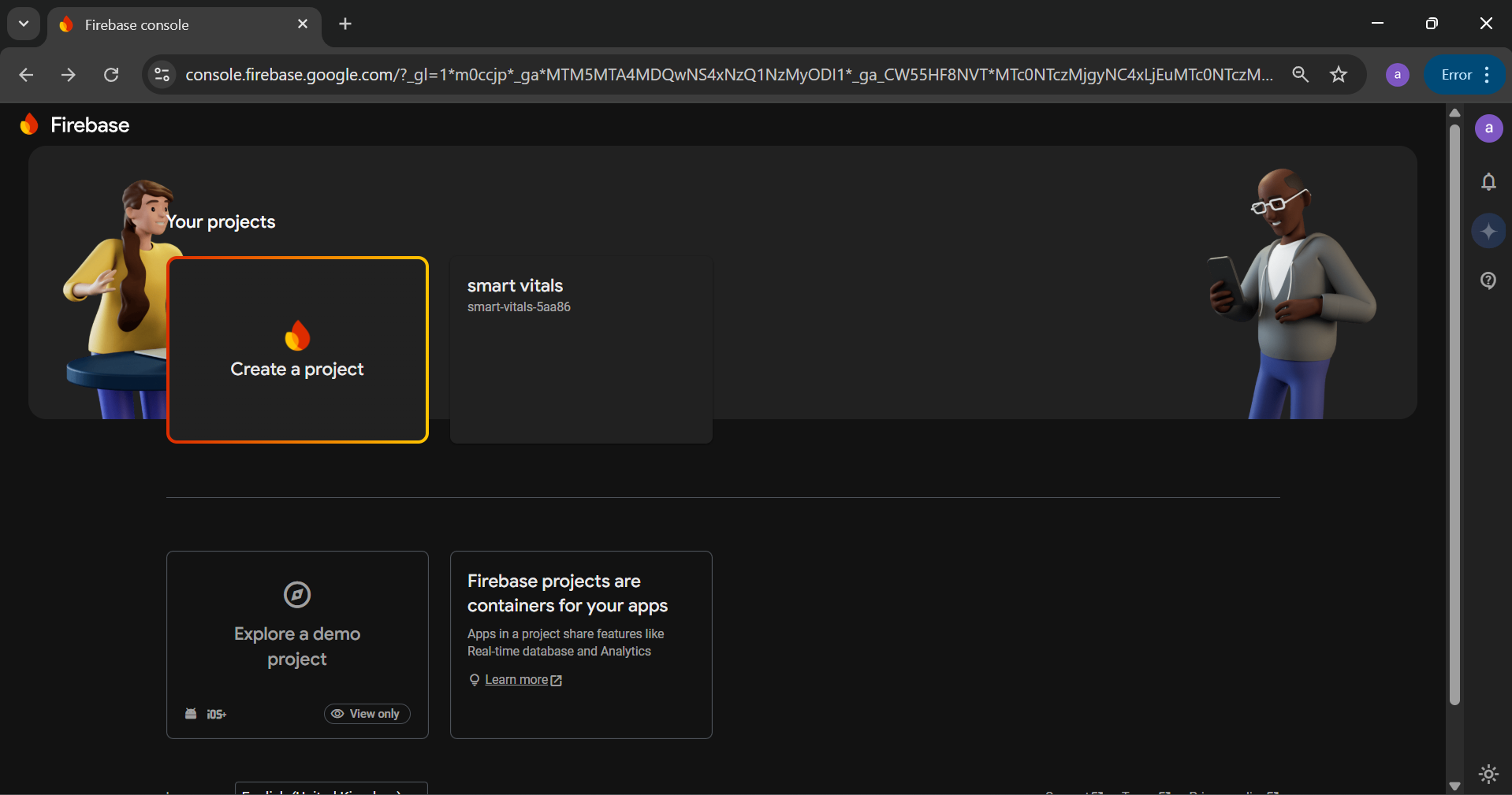
**Step 1:** Before beginning with the installation of firebase package, it is necessary to create project through firebase console.



**Step 2:** Give an appropriate name for your project, and click on continue by agreeing to the terms and conditions.

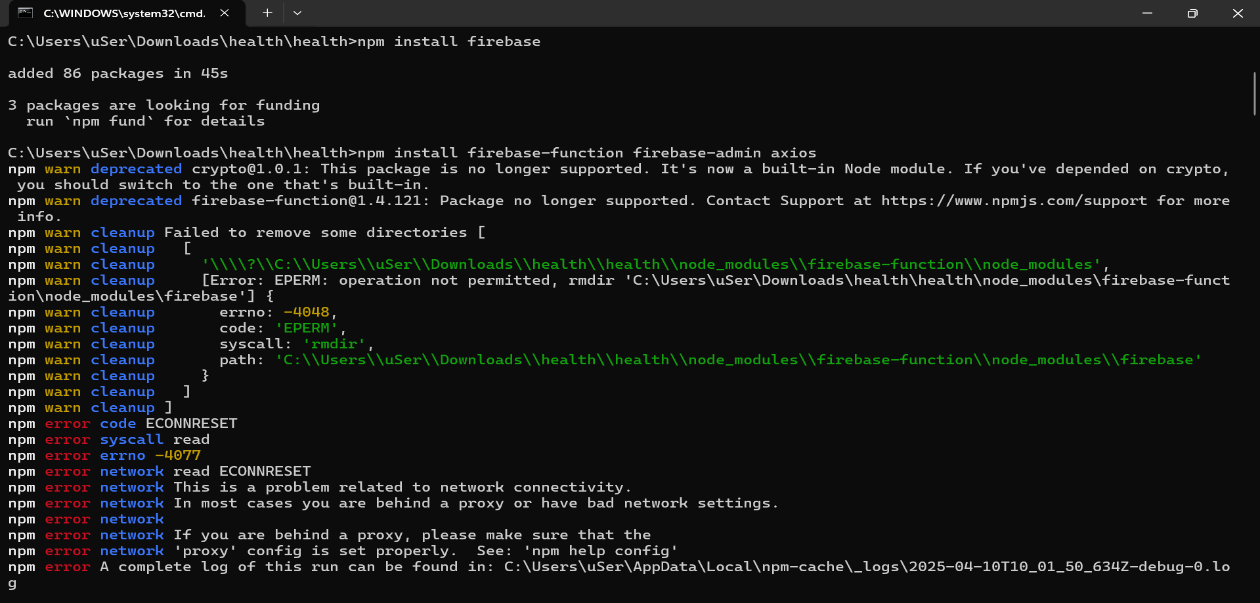
 

**Step3:** The created project can be seen in the firebase console.

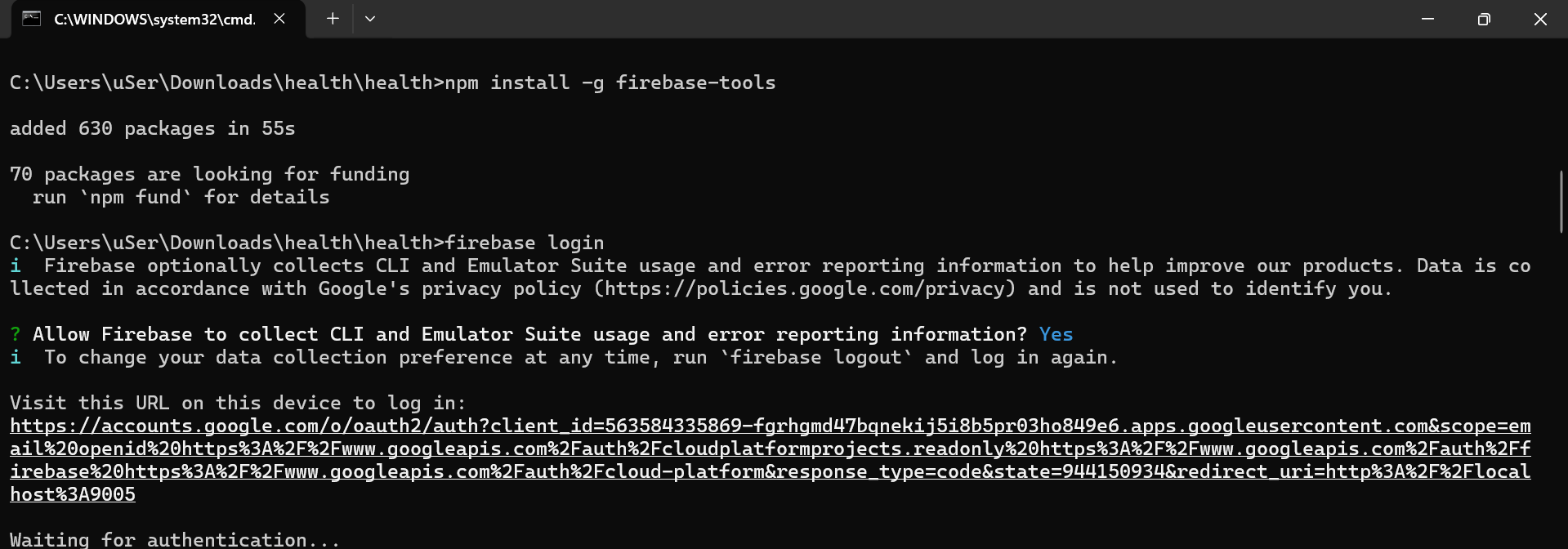


**Step 4:** Once the setup for the firebase console has been done, the installation process can begin.

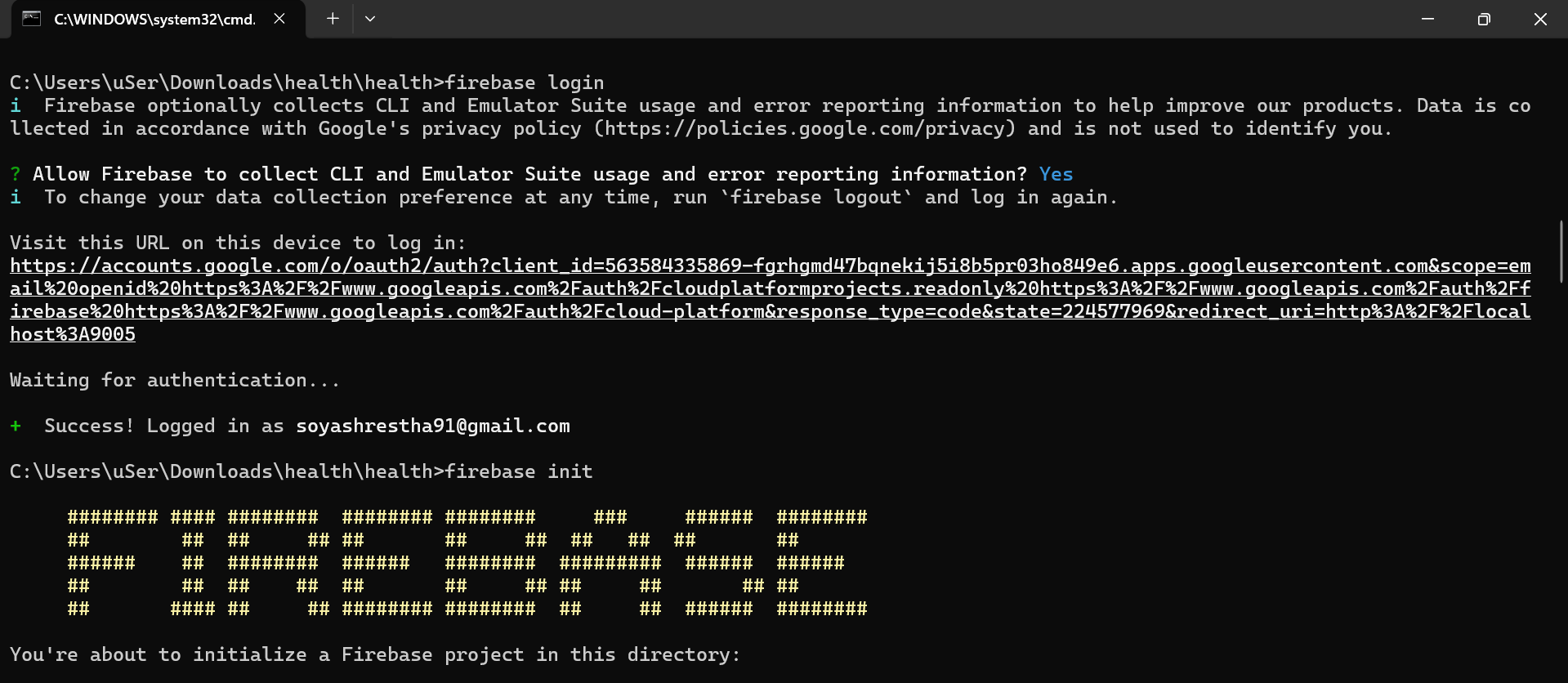
Open cmd on your laptop, and enter “npm install firebase”, this command should install necessary firebase SDK packages.



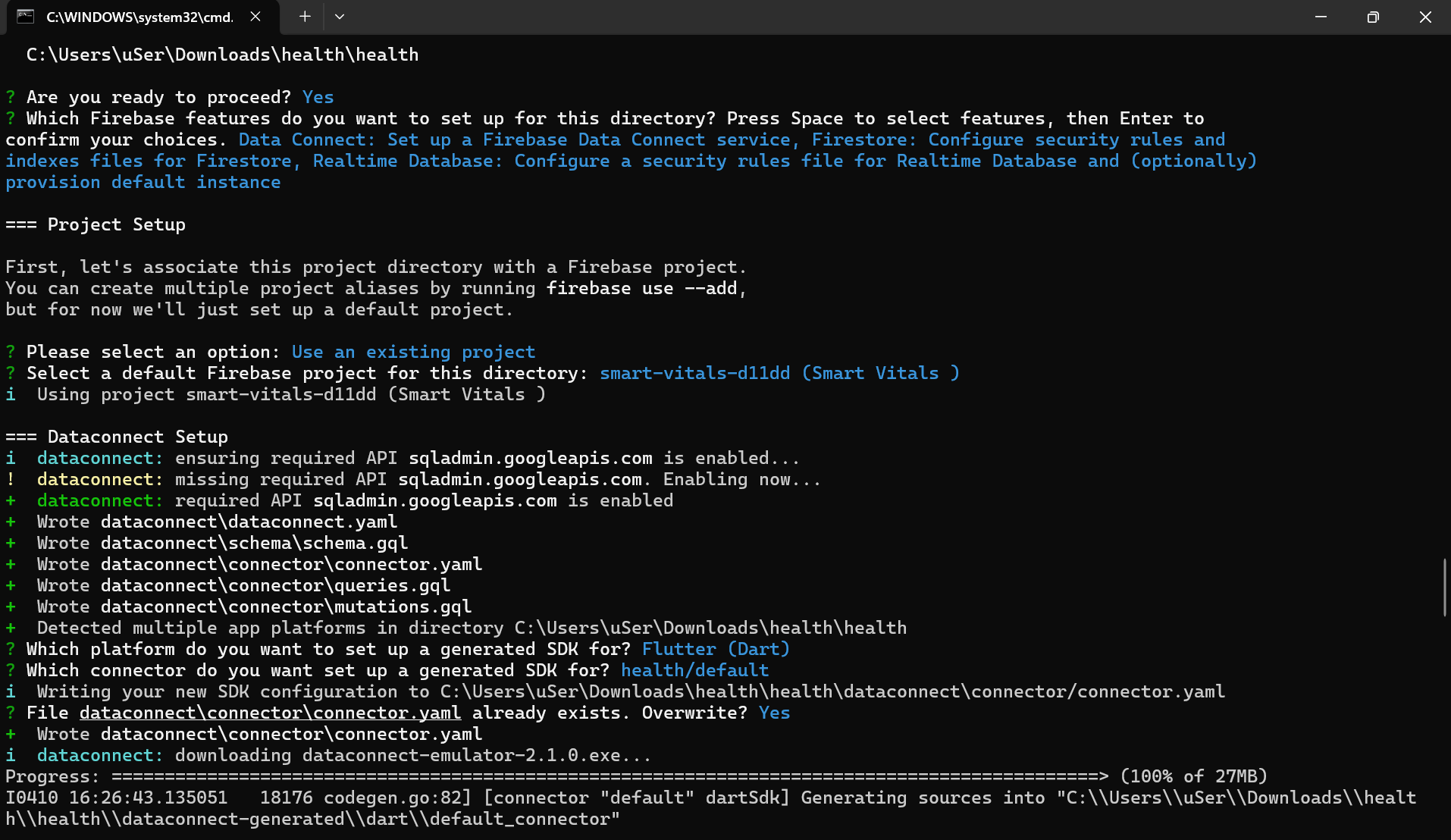
**Step 5:** Next enter “npm install -g firebase-tools” which installs Firebase CLI, which allows us to run global different commands.



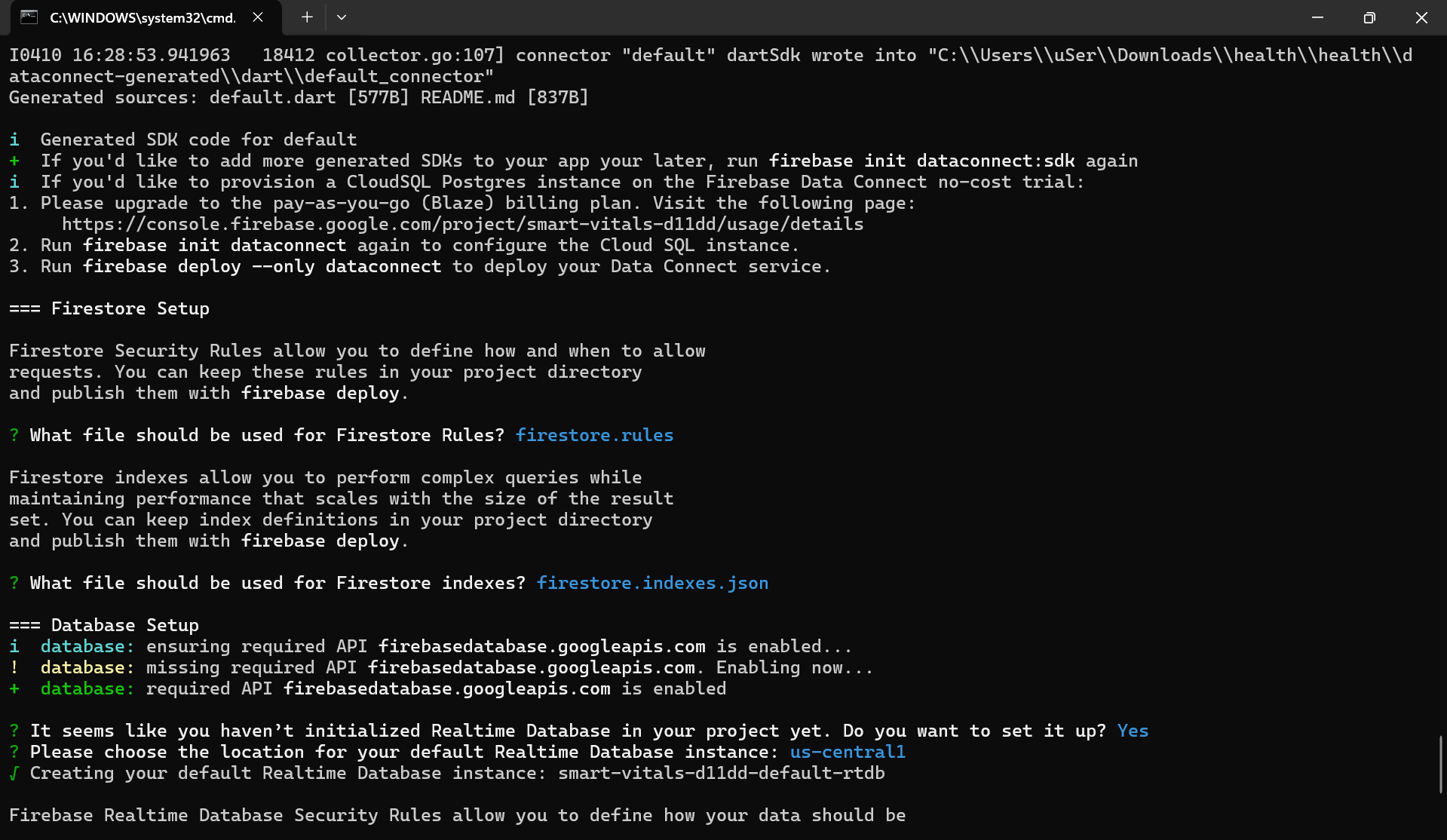
**Step 6**: The command firebase init allows to set up the project folder with the Firebase which allows the user to build, run and deploy the app.



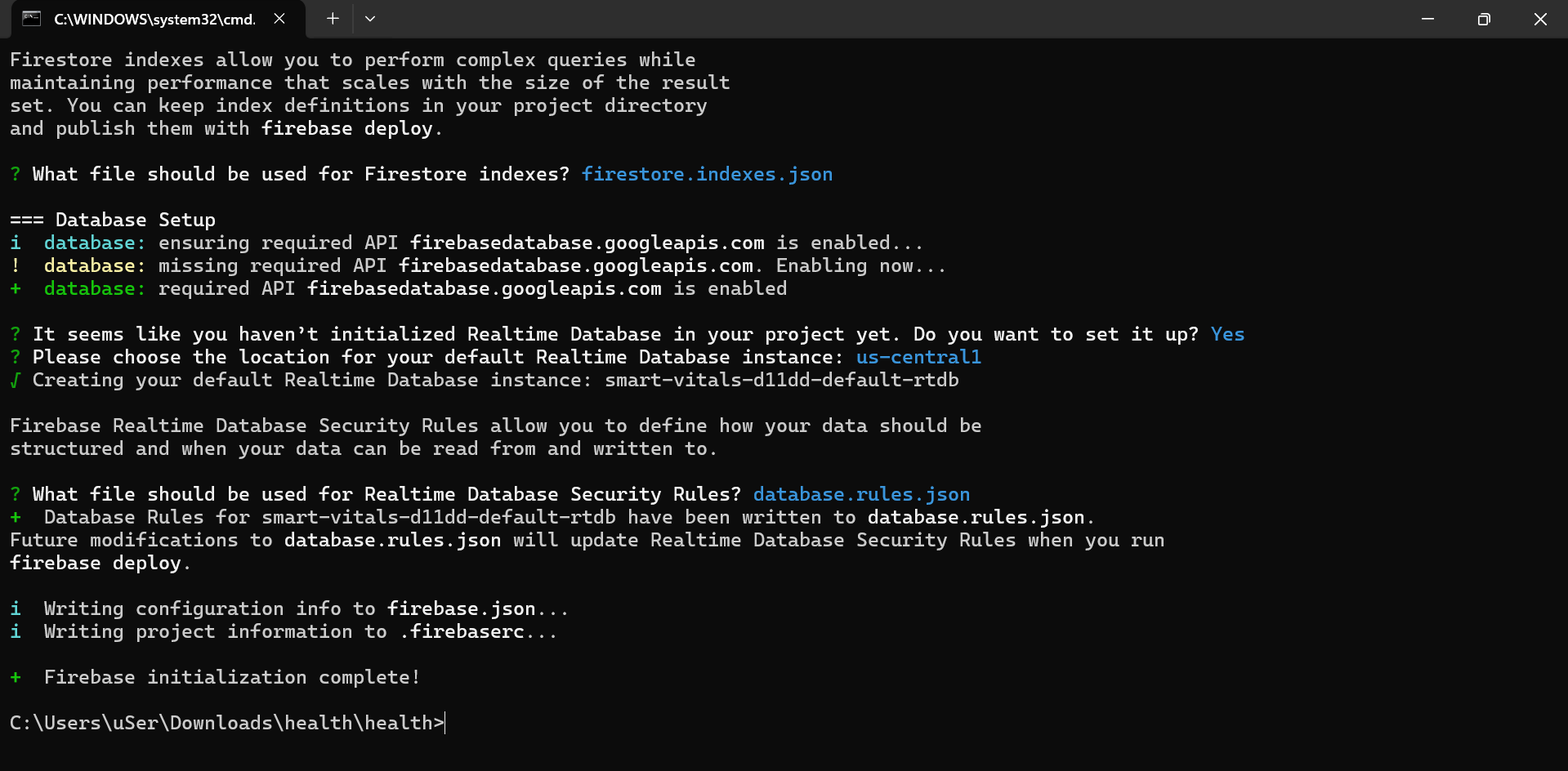
**Step 7:** Then, database schemas, was set up with data connection and then Dart SDK was created which enabled local building and evaluation.



**Step 8:** After that, Firebase indexes and rules which allows API of Realtime Database, creating a new Realtime instance for Database and selecting the location for database instance.

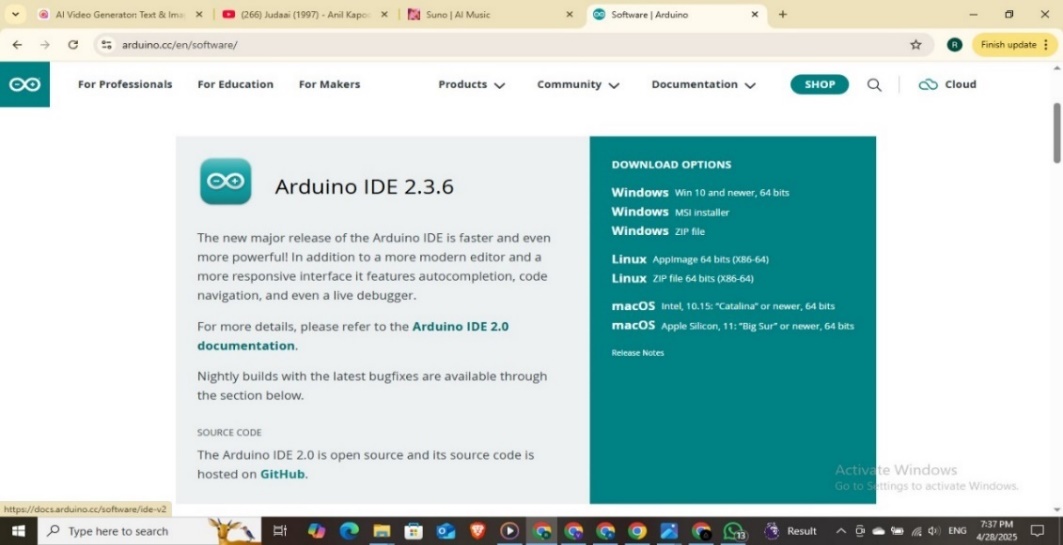


**Step 9**: Then the firebase setup was completed after selecting the file for security rules for Realtime Database.

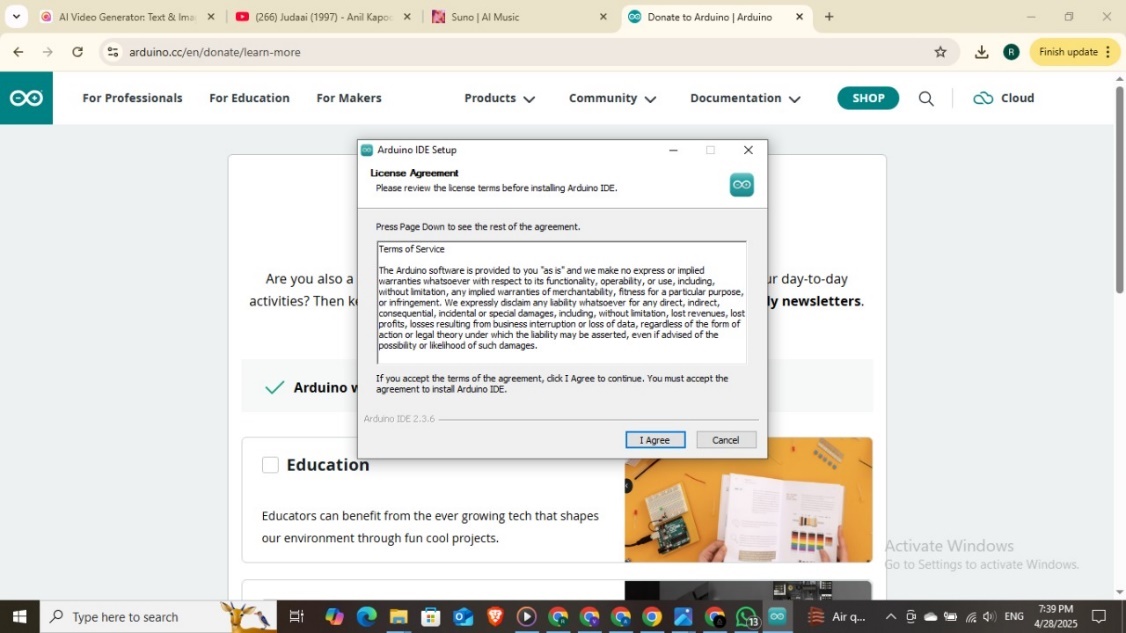


### **9.9.2 Arduino IDE**

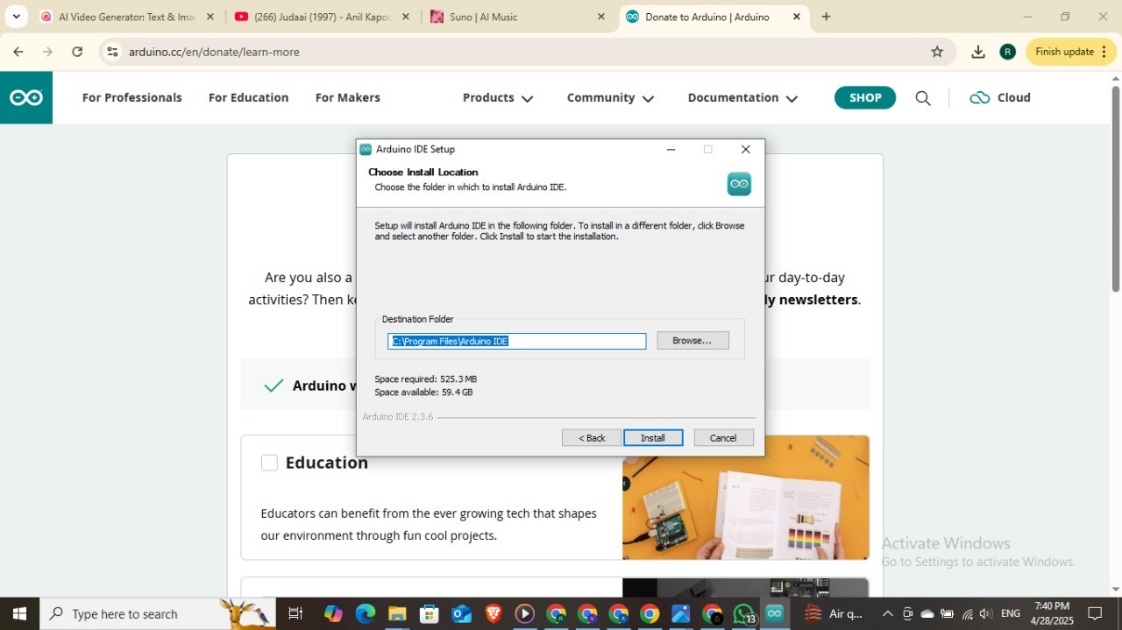
**Step 1:** To install Arduino IDE, we should go to: https://www.arduino.cc/en/software/ where the OS should be selected.



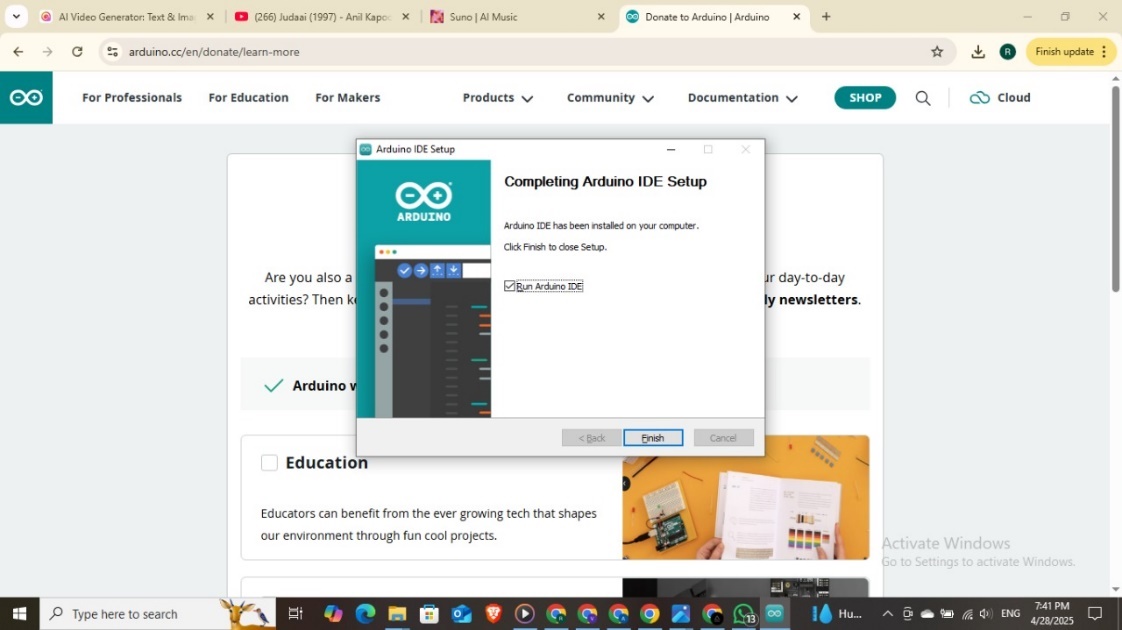
**Step 2:** Then, click “I Agree”, which should start the installation.



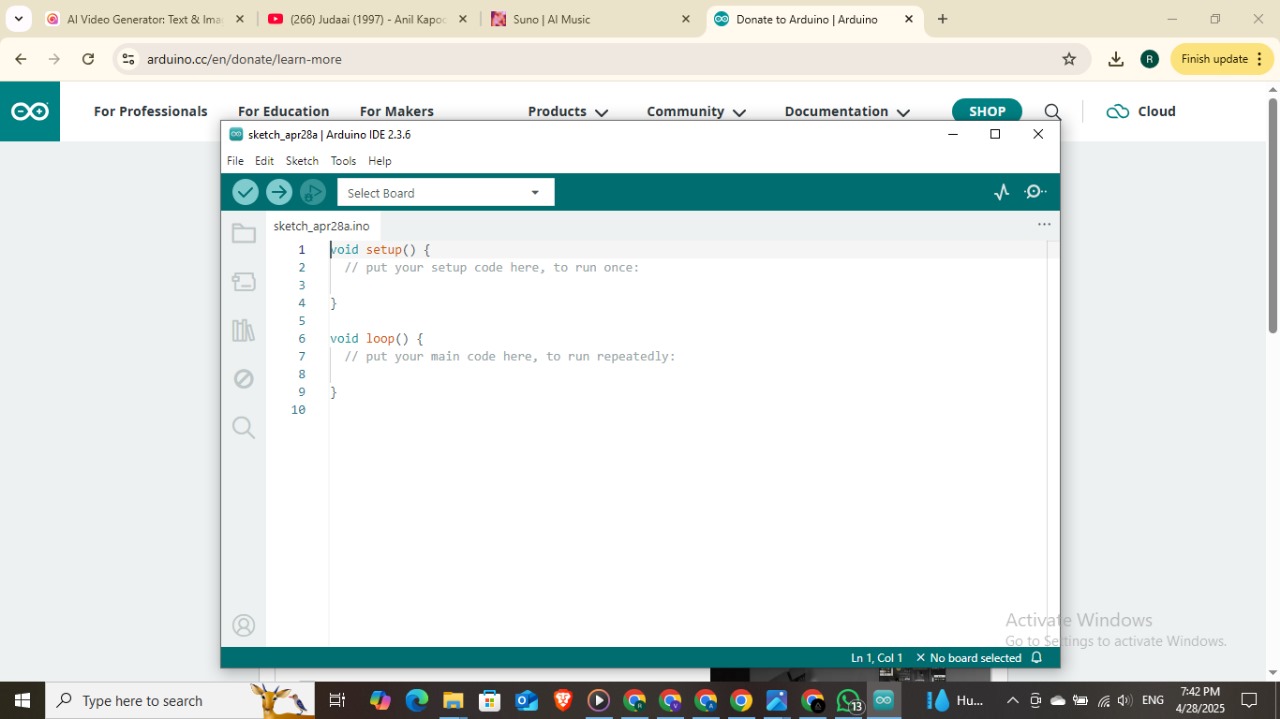
**Step 3:** After that, select the destination folder - where you want to keep your application. Then click on install which should start the installation.



**Step 4**: The setup is completed after the “Completing Arduino IDE Setup” screen is seen as below. Click on Finish to run Arduino IDE.

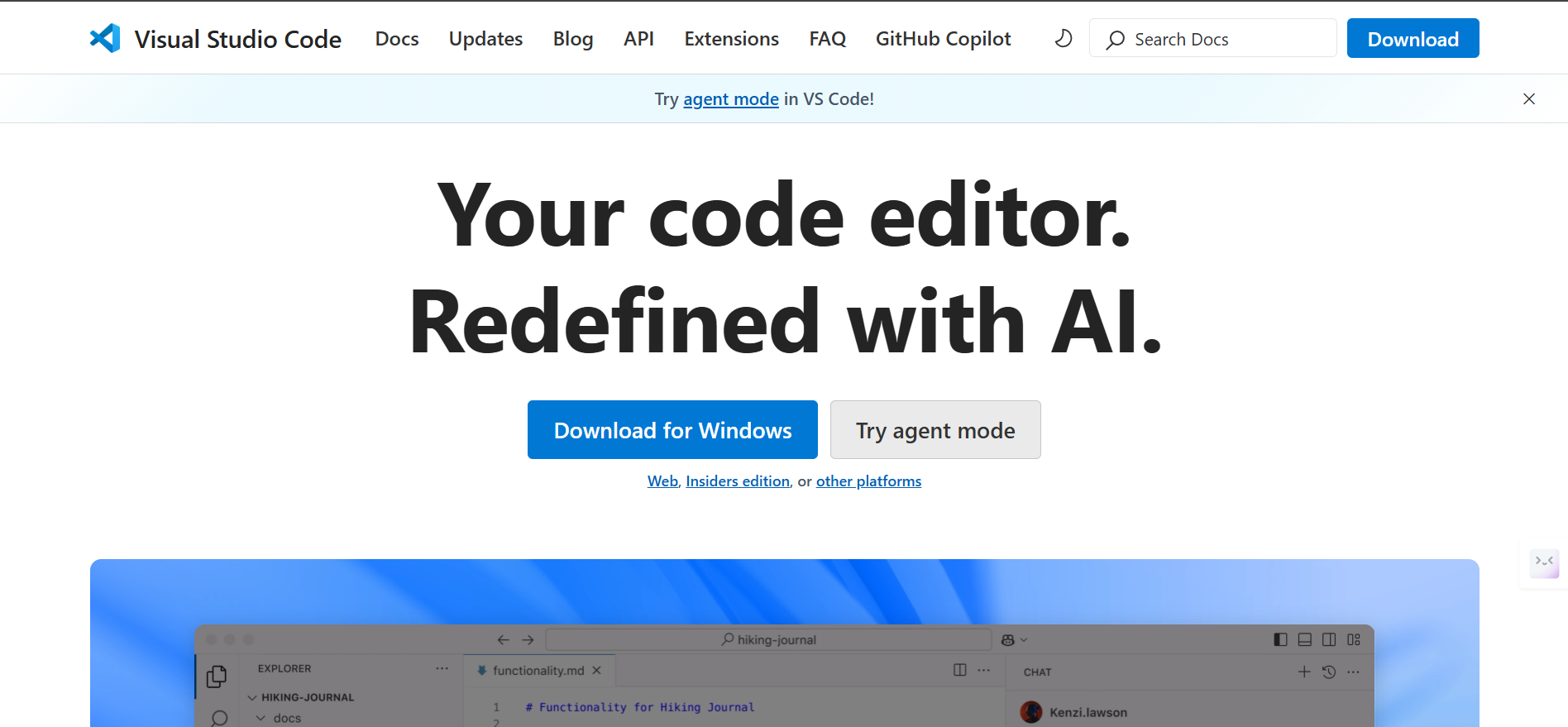


**Step 5:** As you can see, the Arduino IDE is setup and now you start coding.

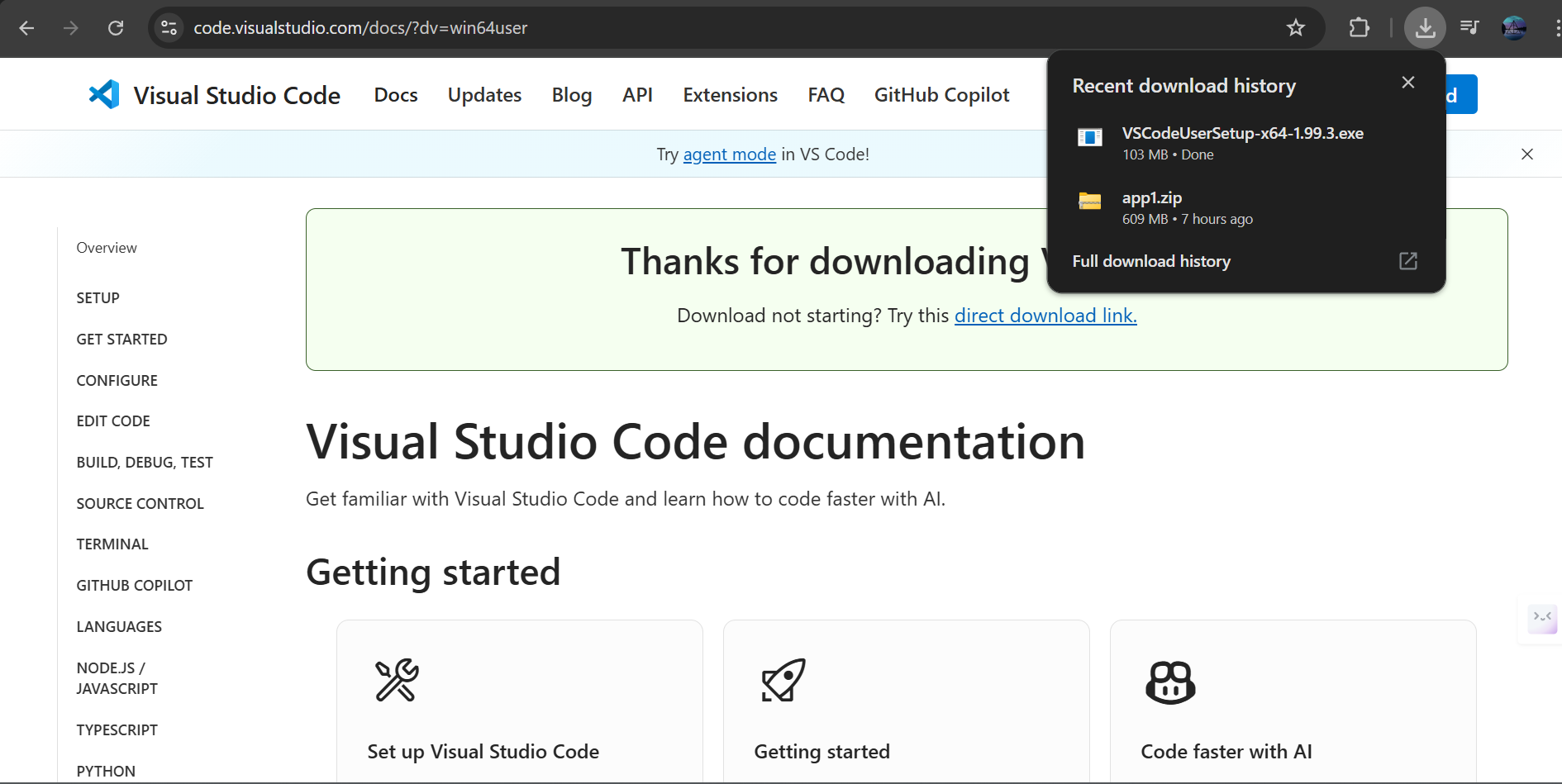


### **9.9.3 VS code**

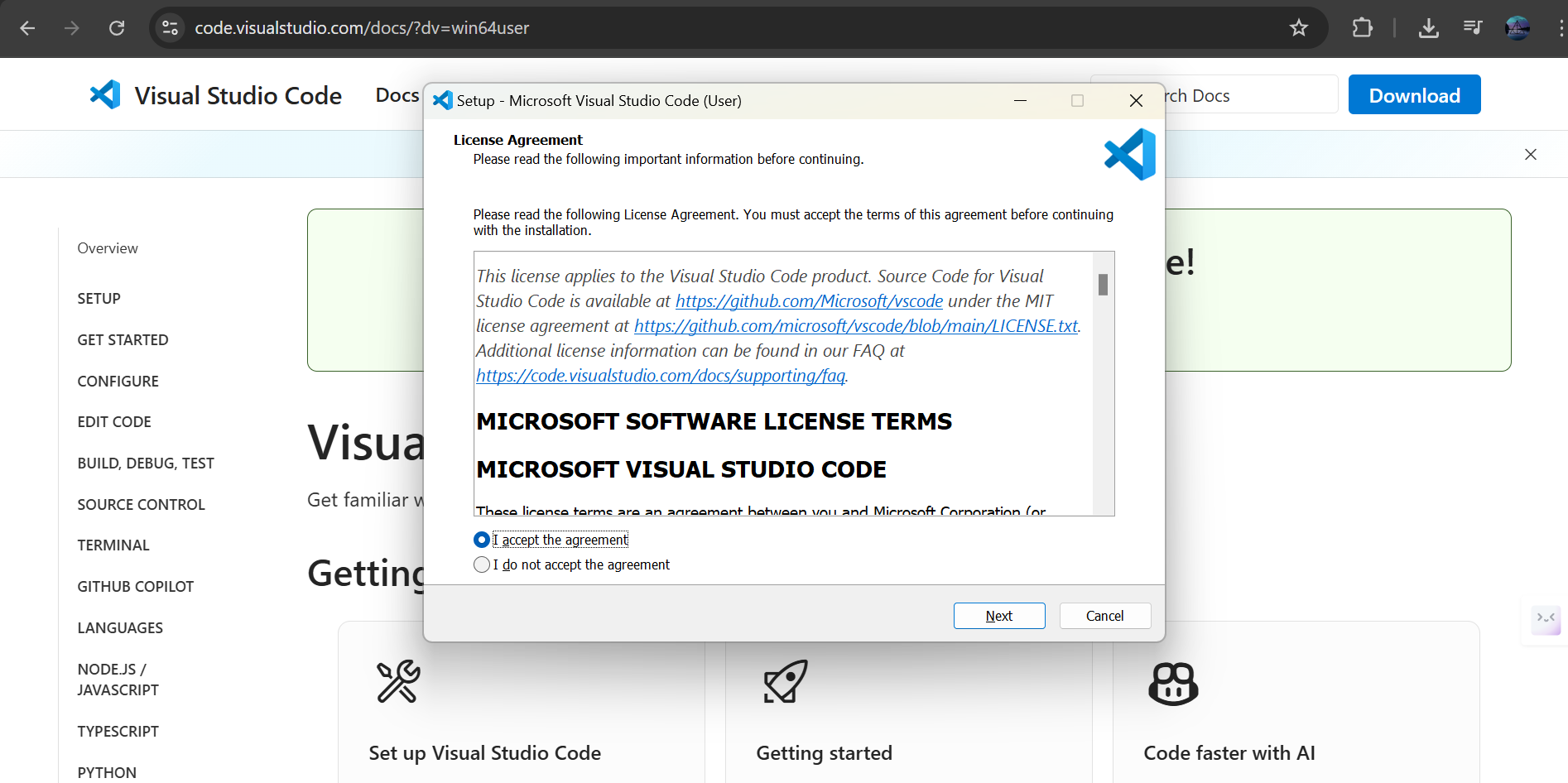
**Step 1:** Go *to* [*https://code.visualstudio.com/*](https://code.visualstudio.com/). Then click on Download for Windows.



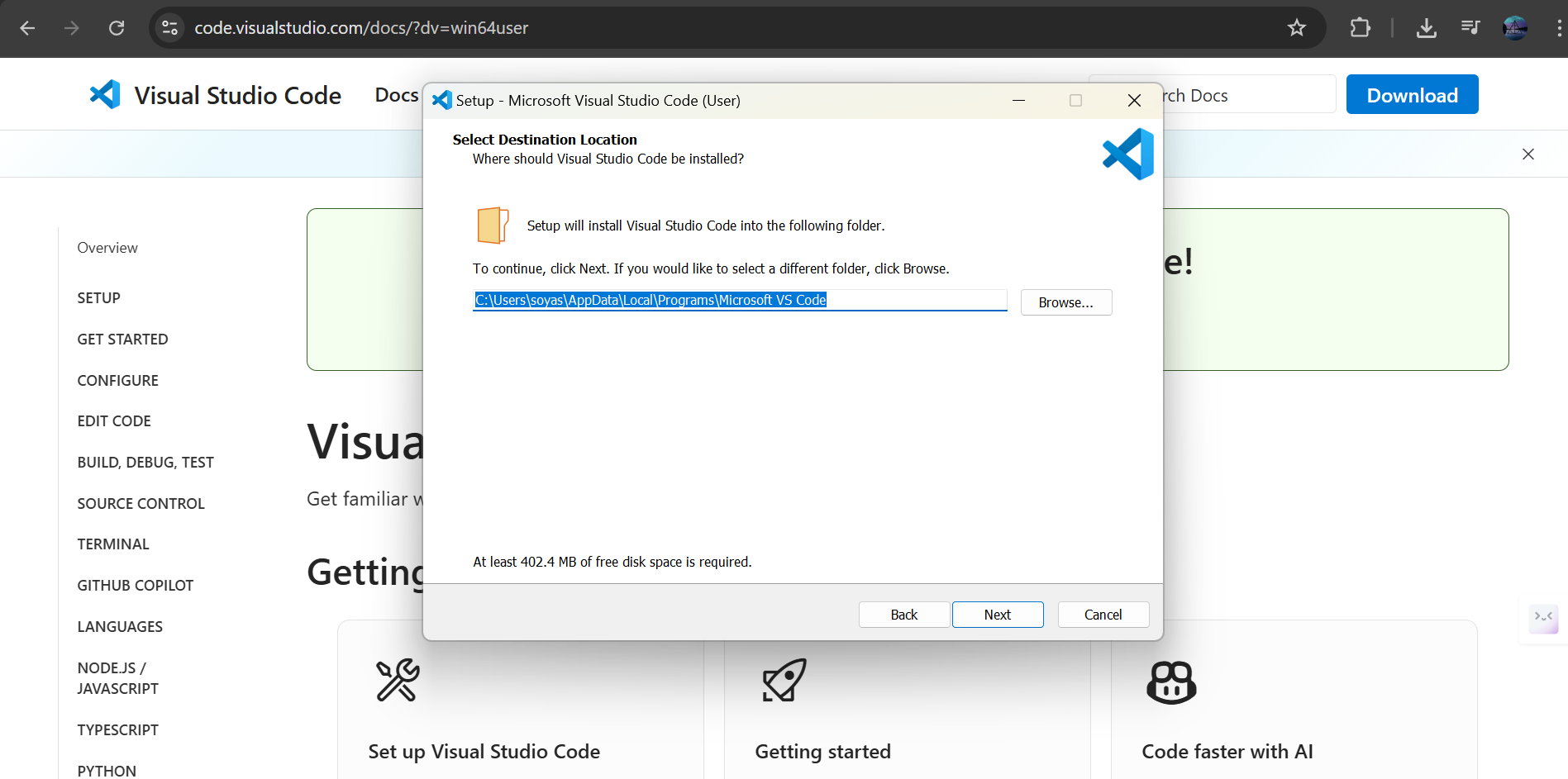
**Step 2:** The setup file should be downloaded as shown in the picture below.

****

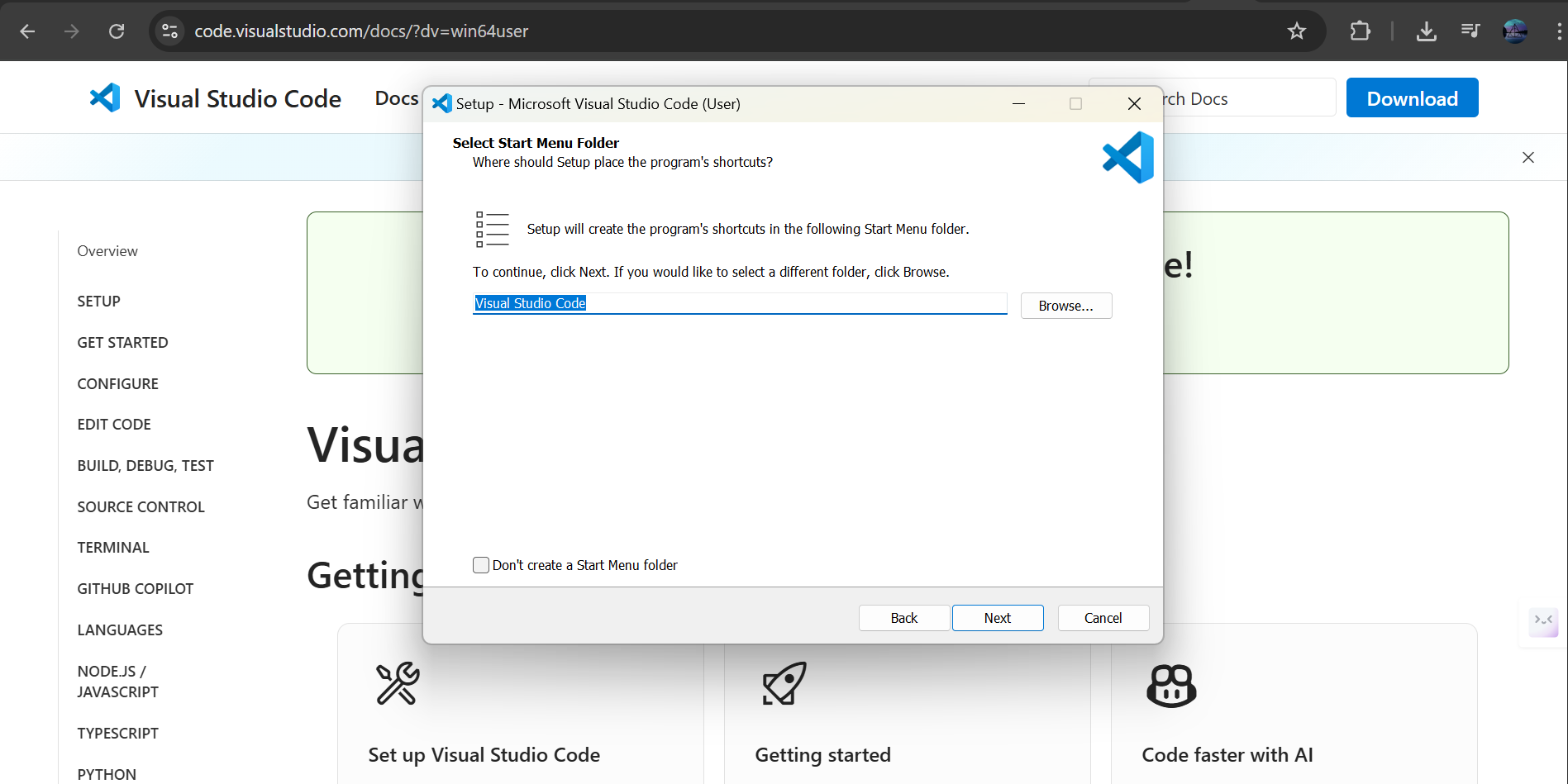
**Step 3:** Now click on “I accept the agreement” and then on Next button.

****

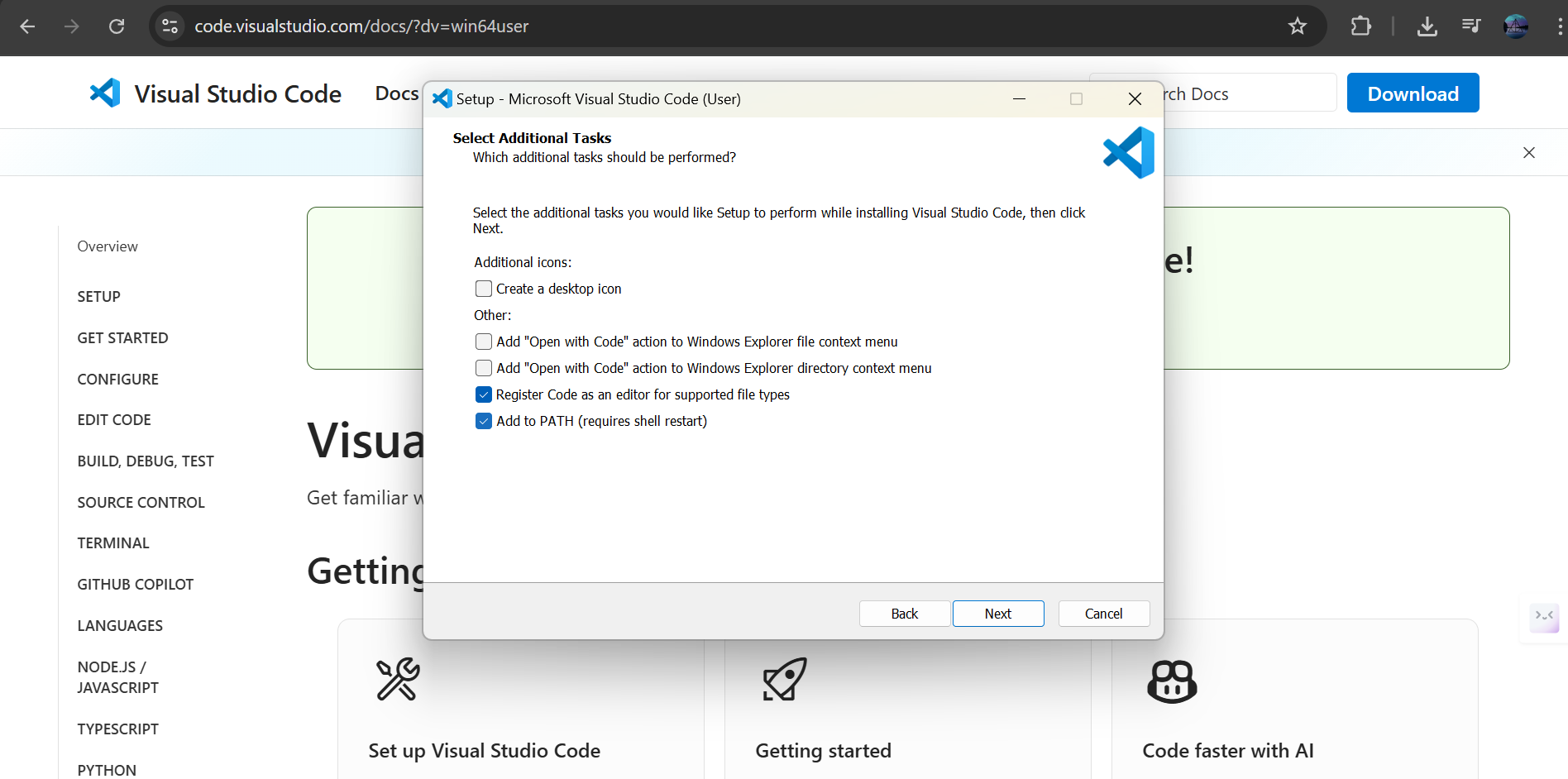
**Step 4:** Select the file destination according to your desire and then click on Next button.

****

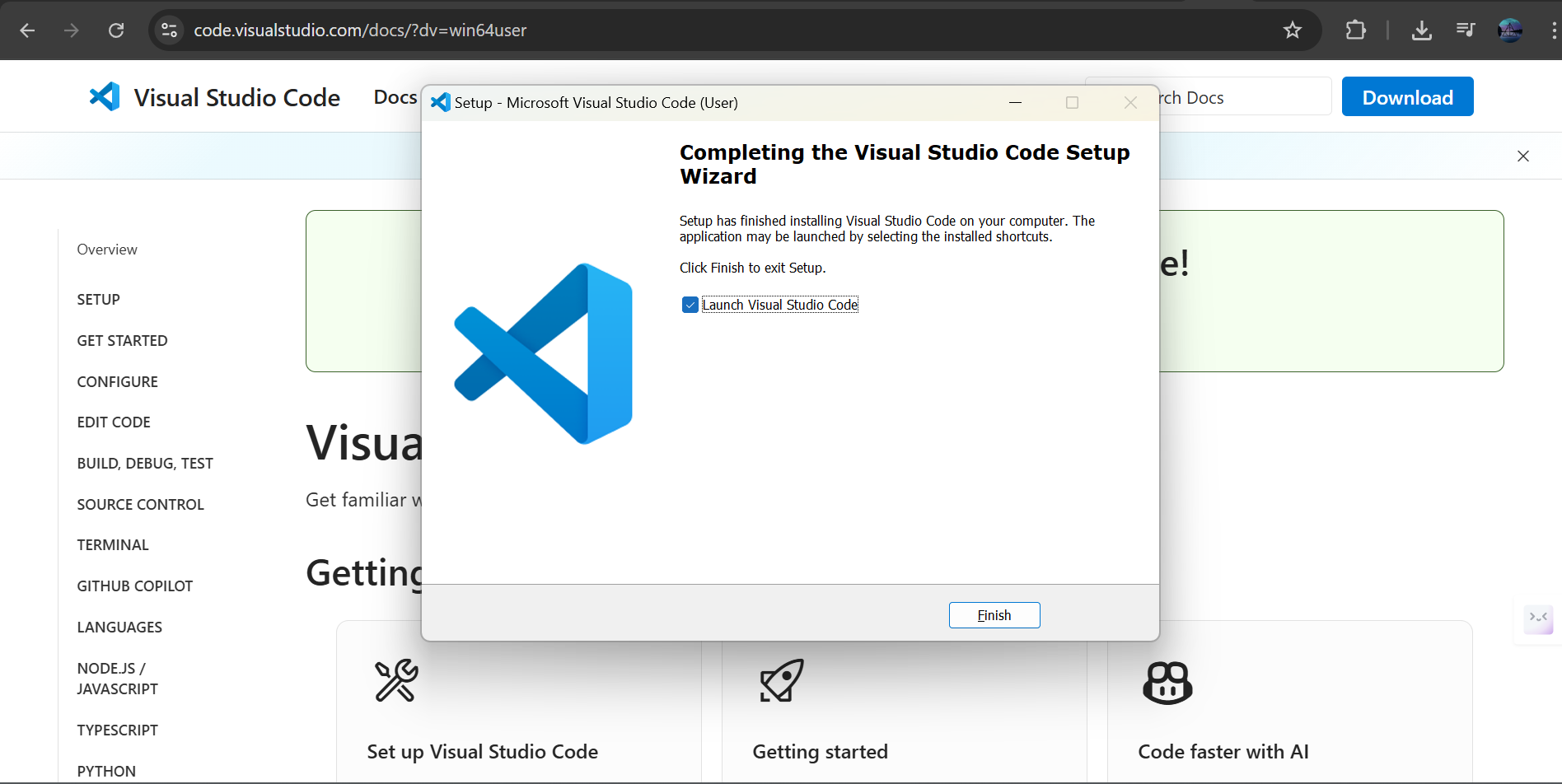
**Step 5:** Click on Next to continue.

****

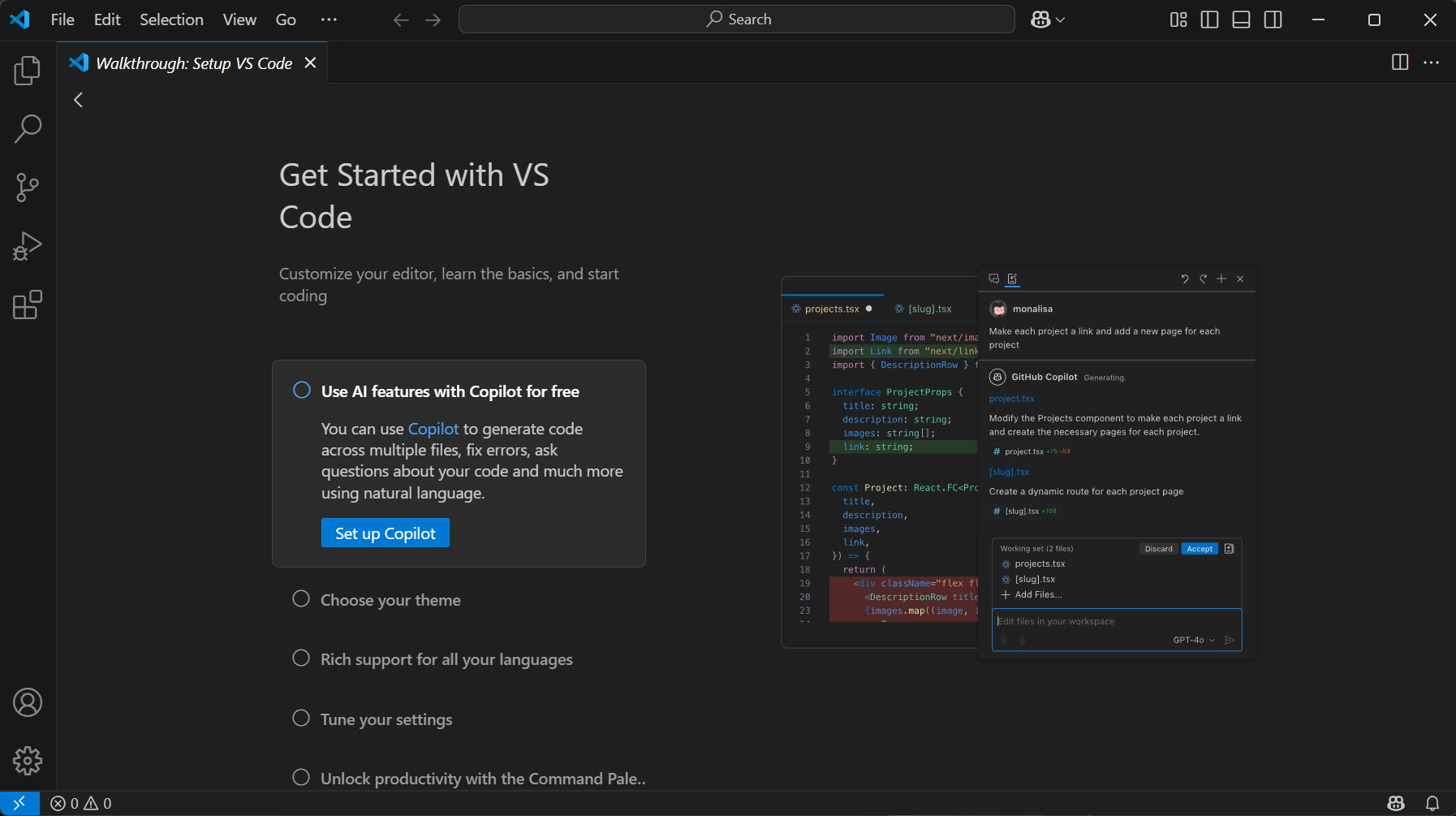
**Step 6:** Leave the default selection or select the additional tasks and then click on Next.

****

**Step 7:** The setup for Visual Studio is now completed, click on Finish to get started with VS code.

****

**Step 8:** You can now start with the coding.

****

## **9.10 Use Case Diagram**



Figure : User Use Case Diagram (1)

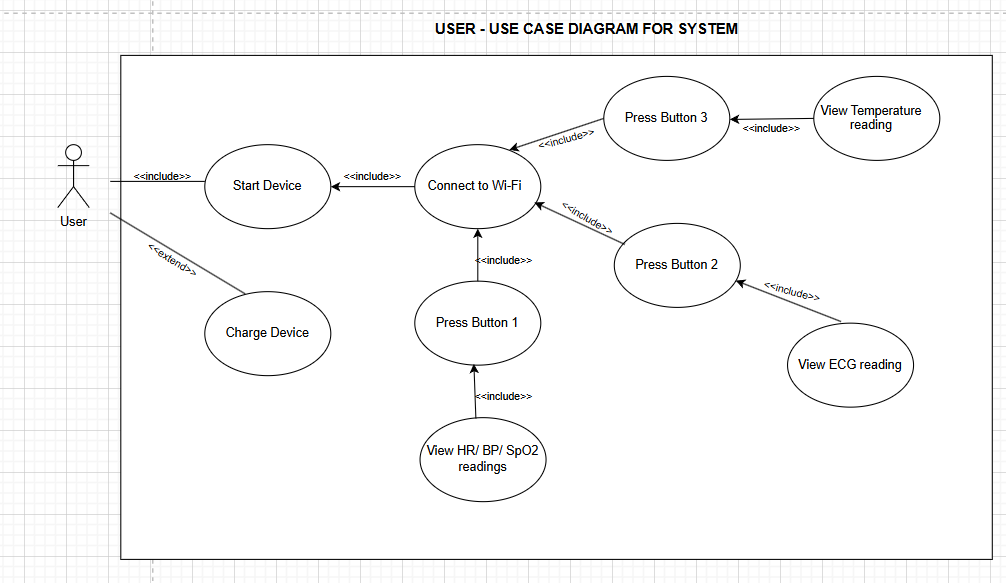


Figure : User Use Case Diagram (2)

The above use case diagram represents how users are associated with the Smart Vitals system and mobile application giving a perspective of the operations inside the system boundary.

## **9.11 Circuit Diagram**

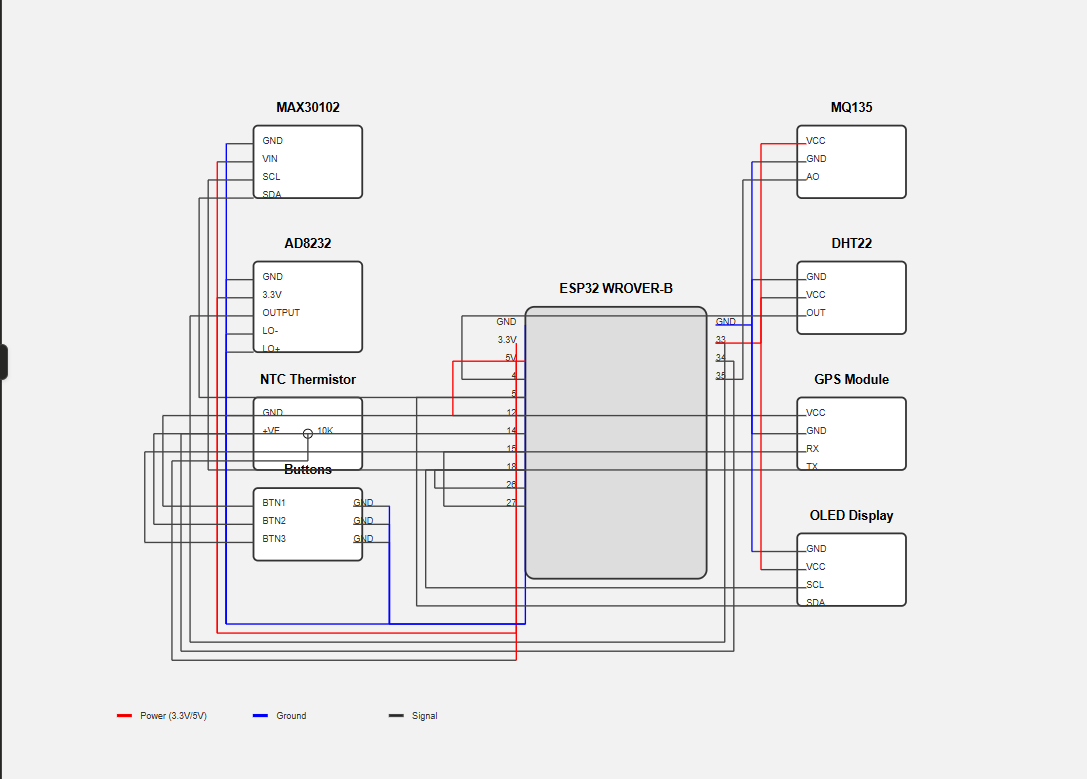


Figure : Circuit Diagram for Smart Vitals

The above circuit diagram represents how each component is associated with the micro controller (ESP 32 Wrover Type B). The positive power supply is represented by the red color wire connections, blue is represented by ground wire (negative power supply) similarly black wire connection is the signal or output connection.

# **CHAPTER 10: SUMMARY AND CONCLUSION**

Smart Vitals was developed to help people get insights on their health conditions. The main focus of the project was on improving health management and easy accessibility to the vitals. It is portable vitals monitoring system that helps users to measure their vitals. The device was designed in such a way that the users can easily get their required measurements with the integration of buttons. If there is a stable internet connection the measurements can be accessible via mobile app however OLED has been included in the system for local access, making it easy and quick to gain readings of the vitals. Voice alert has been added to the mobile app making it a smart feature with live location of the device for helping the users to take instant action. As an additional feature for Smart Vitals, air quality and temperature monitoring has been added to get insights on the environment conditions. The app allows users to save the vital readings according to their need, which could be necessary for further examination allowing the doctors or caretakers to keep track and gain knowledge on their past health history. While testing the product, all the necessary and important features were found to be decent. After doing several testing it was found that the Temperature, using 10k NTC thermistor thermometer and SpO2 using MAX30102 was 99% accurate. While Heart rate, using MAX30102 was 95% accurate. Similarly, the blood pressure was 97% accurate. Further experiments were conducted to test the alert feature, it was found that voice alert was working just fine. Overall, the device performed fair enough, making it a viable product for commercial market on further research and development.

There are many existing products in the market that are somewhat similar to Smart Vitals, while it is still a work in progress device, it has some features that stands out. As per (Valiaugaitė, 2025), Withings BeamO is a portable 4 in 1 device that measures the temperature, blood oxygen level, heart activities and respiratory system. While BeamO is device specifically designed for user health monitoring, Smart Vitals not only measures users’ vitals but also the environments conditions like air quality and temperature. The device is integrated with GPS module so that user can easily get access to the location in case of emergency, with stable Wi-Fi connection. Smart Vitals can give users an estimated measurement for blood pressure while BeamO lacks the feature. BeamO is a high-end premium product which costs around $249.95 according to (Bouchard, 2024), which might not be economical device for the users who are looking for cost effective device, however smartvitals on the other hand is cost effective providing users with more affordable solution. SmartVitals is still a prototype device, additional sensors could be added for more health readings, whereas BeamO is fully developed product thus, additions of more sensors are hardly possible.

Smart Vitals is an example of outcome of technology and health together making it a prototype device for promoting those sectors. Since it is just at a development stage i.e. a work in progress model it could be made into a market-oriented product using more advance and top-tier sensors. To take Smart Vitals to that level several changes could be made. Inclusion of additional sensors such as Blood Glucose Sensor, Electromyography, Stress Level Sensors, etc. could be added for additional vital monitoring. Additionally, further advanced analysis of ECG could potentially be a good additional to the device. Besides that, in the device battery could be further optimized for enhanced battery life. Smart Vitals does have voice alert, but inclusion of real-time alert with notification could be helpful for users to gain knowledge on the users’ irregularities on health vitals. While smartvitals has an accuracy rate of around 96%, it could be improved with the usage of different sensors, more in-depth study and proper calibrations. The usage of AI in the system would be a great way to continue the smart vital project in future as it would help users to predict the health conditions and analyze vital trends. While these are few examples of how smart vitals can be upgraded in the future, there are many other ways to enhance the system making it compatible with other exiting products in the market.